TURING 图灵电子与电气工程丛书

McGraw-Hill Education

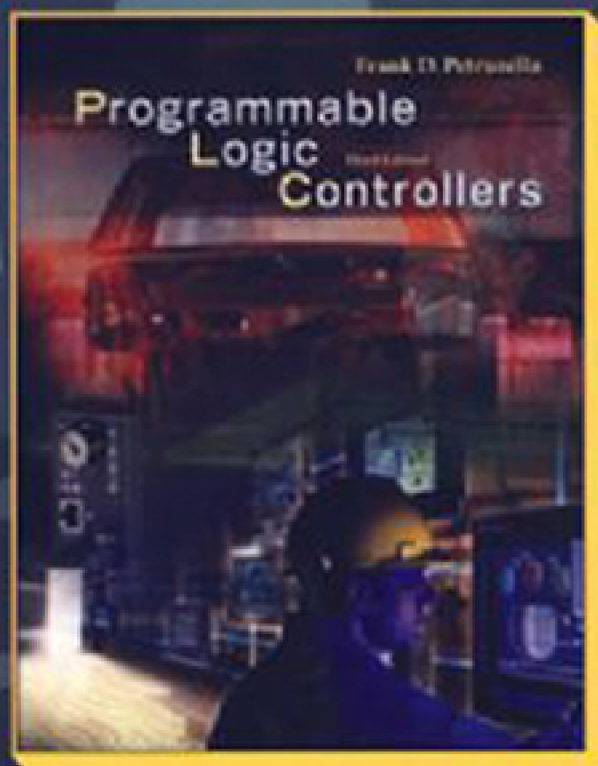
PLC教程

（第3版）

Programmable Logic Controllers

Third Edition

[美] Frank D. Petruzella 著
侯世英 等译





人民邮电出版社
POSTS & TELECOM PRESS

TURING 图灵电子与电气工程丛书

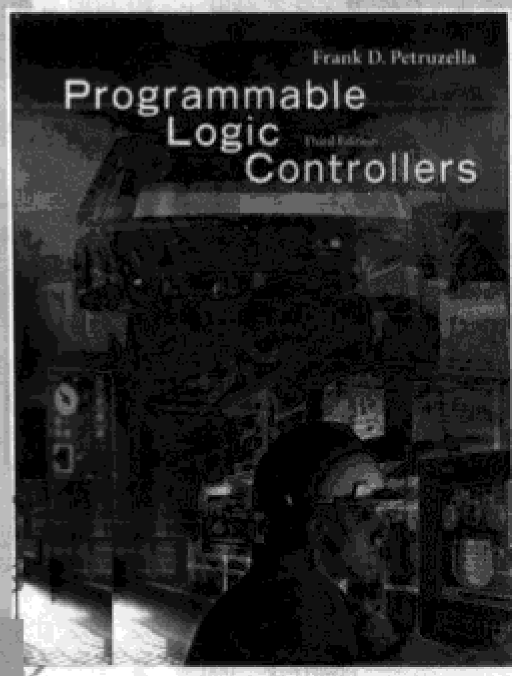
PLC教程

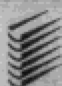
（第3版）

Programmable Logic Controllers

Third Edition

［美］Frank D. Petruzella 著
侯世英 等译





人民邮电出版社
POSTS & TELECOM PRESS

图书在版编目（CIP）数据

PLC 教程（第 3 版）/（美）皮特鲁泽拉（Petruzella，F. D.）著；侯世英等译.
—北京：人民邮电出版社，2007. 5
（图灵电子与电气工程丛书）

ISBN 978-7-115-15889-5

Ⅰ. P…　Ⅱ. ①皮…②侯…　Ⅲ. 可编程序控制器　Ⅳ. TP332. 3

中国版本图书馆 CIP 数据核字（2007）第 025034 号

内 容 提 要

本书是一本在国外广受欢迎的 PLC 教材。本书从 PLC 的共性特征出发，重点讲述 PLC 的基本工作原理，并提供如何对 PLC 控制系统进行安装、设计和维护的实用信息。本书内容通俗易懂，配有大量清晰的图表，便于学生理解 PLC 系统的基本原理；每一章后面都配有与该章学习目标紧密相关的思考题和习题，便于学生巩固所学内容。

本书适合作为大学本科电子类及相关专业的 PLC 教材，也适合从事 PLC 应用和开发的工程技术人员。

图灵电子与电气工程丛书

PLC 教程（第 3 版）

◆ 著　［美］Frank D. Petruzella
译　侯世英　等
责任编辑　朱　巍

◆ 人民邮电出版社出版发行　北京市崇文区夕照寺街 14 号
邮编　100061　电子函件　315@ptpress.com.cn
网址　http://www.ptpress.com.cn
北京铭成印刷有限公司印刷
新华书店总店北京发行所经销

◆ 开本：700 × 1000　1/16
印张：20.25
字数：540 千字　2007 年 5 月第 1 版
印数：1－5 000 册　2007 年 5 月北京第 1 次印刷

著作权合同登记号　图字：01-2006-3692 号

ISBN 978-7-115-15889-5/TN

定价：45.00 元

读者服务热线：(010)88593802　印装质量热线：(010)67129223

前　言

可编程序控制器（PLC）广泛应用于工业领域的各个方面，以扩大生产规模并提高产品质量。一个经过编程的PLC，可以替代一些旧自动控制系统中所需的成百上千个继电器。经过多年的发展，PLC的功能已经超出了传统的继电控制，它已经延伸到高精度运动控制系统、过程控制系统和分散控制系统，并具备了复杂的通信联网功能。

在本书最新的第3版中，我们介绍了PLC的全部性能，包括其运行操作和在各方面的应用；重点讲述了PLC的基本工作原理，并提供了如何对PLC控制系统进行安装、设计和维护的实用信息。本书不需要读者先前了解PLC和具备编程方面的知识。

对于实际的PLC机型，主要的信息参考资料来源都是由供应商提供的用户手册。本书并不想取代用户使用手册，而是要作为手册内容的补充、阐述和扩展。现今生产PLC的厂家很多，因此在一本书中要涵盖PLC不同的制造商和机型的细节是不切实际的。基于这种思想，本书把PLC作为一类，仅讨论通用的情况。尽管书的内容适用于各种型号的PLC，在本书的编程实例中采用的是广受欢迎的Allen-Bradley PLC-5、SLC-500和ControlLogix控制器的指令系统。

本书通俗易懂，书中有很多清晰的图表可以帮助学生理解PLC系统的基本原理。每一章前面都有一个学习目标和主要内容提要，便于学习时参考。文中先解释了与继电器电路等效的程序指令，然后再介绍与之对应的PLC指令。每一章的后面都有一组思考题和习题，它们与该章前面的学习目标紧密相关，能够帮助学生评价学习的情况、增进对本章内容的理解。

书中所有的主题都分成小的段落，这样学生就可以在进入下一个主题学习之前，牢固掌握概念和操作，打下坚实的基础。书中有一整章介绍适用于PLC的逻辑电路。PLC使用的安全规则和注意事项贯穿于全书。书中技术术语在第一次使用时，将给出定义，此外通过书后面提供的术语表也可以方便地查询与PLC有关的技术术语。本书还介绍了PLC系统故障诊断的一般步骤，指导学生如何对PLC故障进行系统的分析。

本书在前一版基础上作了修订，包含以下几个新的特征：

- ❑ 对基于RSLogix视窗的编程作了详述。
- ❑ 增加了ControlLogix的编程指令。
- ❑ 所有的章节都作了更新，包含了最新发展状况。
- ❑ 更新了硬件设备表。
- ❑ 扩展了术语表。

致谢

我要对以下审稿人的意见和建议表示感谢！他们是：

Lake Area技术学院（南达科他州沃特敦）的Christine L. Berger-Wilkey

国立技术学院（加州长滩）的Thomas E. Clark

佛罗里达社区学院的 David C. Kier
东南社区学院（西伯灵顿）的 John Lenahan
Computer Electronics Technology 公司的 William Salice
约翰逊县社区学院的 David A. Setser
SCP Global Technologies 公司的 Dan Siddall

另外，我要感谢收到本书写作期间发出的调查函并回复的每一个人。最后我要感谢 McGraw-Hill 教育出版公司的 David Culverwell、Pat Forrest、Jane Mohr、Karen Dorman 和 David Tietz，感谢他们在本书的出版过程中给予的帮助。

第 1 章　PLC 概述

学习目标：

- ❑ 定义 PLC 并且列出它与继电器系统相比的优越性。
- ❑ 描述组成 PLC 的各主要部分及功能。
- ❑ 描述 PLC 基本的工作过程。
- ❑ 确定 PLC 的基本分类。

本章讲述了 PLC 的产生及发展历史，讨论了从继电器控制系统到 PLC 系统变化的原因。从构成 PLC 的基本部分入手，介绍了 PLC 在过程控制系统中的应用，然后简单介绍了各种不同类型的 PLC 的特点及应用情况，最后介绍了一种可以简化 PLC 控制的编程语言——梯形图语言。

1～2

1.1　PLC

如图 1-1 所示，PLC 是一种用于控制生产机器和工作过程的特殊计算机。它使用可编程序的存储器存放指令，并按照指令执行，完成相应的开/关控制、定时、计数、顺序控制、算术运算和数据处理等功能。

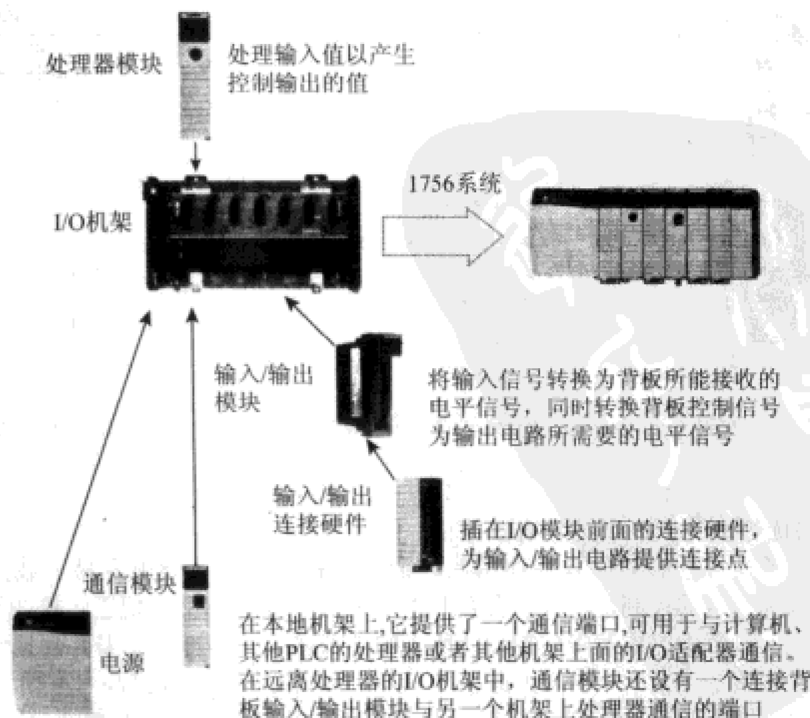


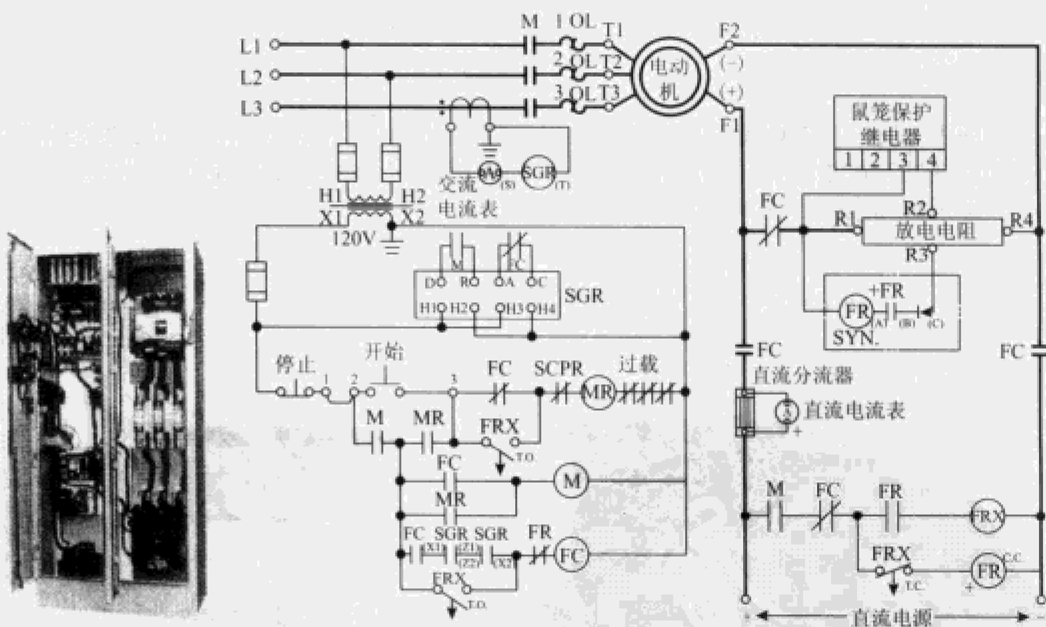
图 1-1　PLC（得到 Rockwell 公司许可）

通常，大多数 PLC 的设计与计算机相似，其内部为一系列固态的数字逻辑元件按照一定顺序排列，来完成逻辑运算，并提供输出结果，从而完成控制生产设备和机械的任务。

因此，PLC 是一种专门用于机器控制的计算机，与办公用计算机不同的是，它采用特殊的输入/输出接口电路和逻辑控制语言，以在工业环境中工作。工业界将这类设备简称为 PC（Programmable Controller），但是这个简称容易与“个人计算机（Personal Computer）”的缩写 PC 混淆，所以，部分生产商将其简称为 PLC，它是 Programmable Logic Controller 的首字母缩写。

最初，PLC 只是用于取代继电器控制系统中的逻辑控制，但是因为 PLC 的指令采用了与计算机算术指令相同的原理，使其应用范围和复杂性不断增加，使其不仅能够完成继电器的开关任务，而且可以完成计数、运算、比较和模拟信号处理等任务。

PLC 与传统的继电器控制相比具有一系列优点。继电器控制系统必须有硬接线才能完成特定的功能，如图 1-2 所示。当系统控制需求改变时，继电器控制电路的接线也必须作相应的改变或者改进。在极端的情况下，例如在汽车业中，必须换掉整个控制板，因为每次型号升级时都要改接线路，这在经济是不可行的。PLC 避免了传统继电器电路大量的硬接线，与基于继电器的过程控制系统相比，它的体积更小，价格也不贵。



(a) 典型的硬接线控制板　　(b) 典型的硬接线原理图（得到Allen-Bradley公司许可）

图 1-2[1]

除了价格优势之外，PLC 还具有以下优点：

- **更高的系统可靠性**。一旦程序编写完成并经过调试，就能很容易下载到其他 PLC 上。因为所有的控制逻辑都包含在 PLC 的存储器中，不会出现逻辑接线错误。用程序取代了过程中完成逻辑控制功能所需的外部接线，尽管仍然需要硬接线连接现场设备，但数量已大大减少。因此，PLC 与一些固态元件能提供高可靠性。
- **更好的系统灵活性**。在 PLC 中编写和修改程序比电路的接线和修改接线容易得多。原始设备生产商只需要简单发布出新的程序，就能实现对系统的升级。终端用户可

1. 本书中各种电路符号均采用原著中的美国电气标准。——编者注

以在现场修改程序，如果需要，还可以提供例如硬件方面的键盘锁或者软件方面的密码等保护功能。

- 更低的系统成本。最初设计 PLC 是为了取代继电器控制逻辑，其节省费用的效果已经非常明显，除了还在电力系统应用以外，继电器控制已经过时。通常，如果一个控制系统中使用了 6 个以上的继电器，用 PLC 代替则费用会更低。
- 通信能力。PLC 能够与其他的控制器或者计算机设备通信以完成监控、数据采集、监视设备和过程参数以及下载、上传程序等功能。
- 更快的系统响应时间。PLC 是为高速和实时应用而设计的。PLC 用于实时控制，指的是现场发生的事件会引起某一操作或输出的执行。每秒处理数千条指令的机器以及识别在传感器前动作时间不足一秒的物体，都需要 PLC 的快速响应能力。
- 更容易的故障诊断。PLC 固有的自诊断和超控功能，使用户很容易地找出并改正软件和硬件的故障。为了查找和确定故障，用户可以把控制程序显示在监视屏上，并在它执行时进行实时观察。

1.2 PLC 的组成部分

一个典型的 PLC 可由图 1-3 所示的各部分组成。它们是中央控制单元（CPU）、输入/输出（I/O）部分、电源和编程器。术语“体系结构”是指 PLC 的硬件，或软件，或者二者的结合。开放的体系结构，是指系统使用现成的标准组件，能方便与其他生产厂家的设备和程序兼容。封闭的体系结构则是指该系统为专用的并且与其他系统不兼容。目前，大多数 PLC 系统从本质上讲都是封闭的系统，所以，使用时必须确定所选用的硬件和软件与所使用的 PLC 是兼容的。

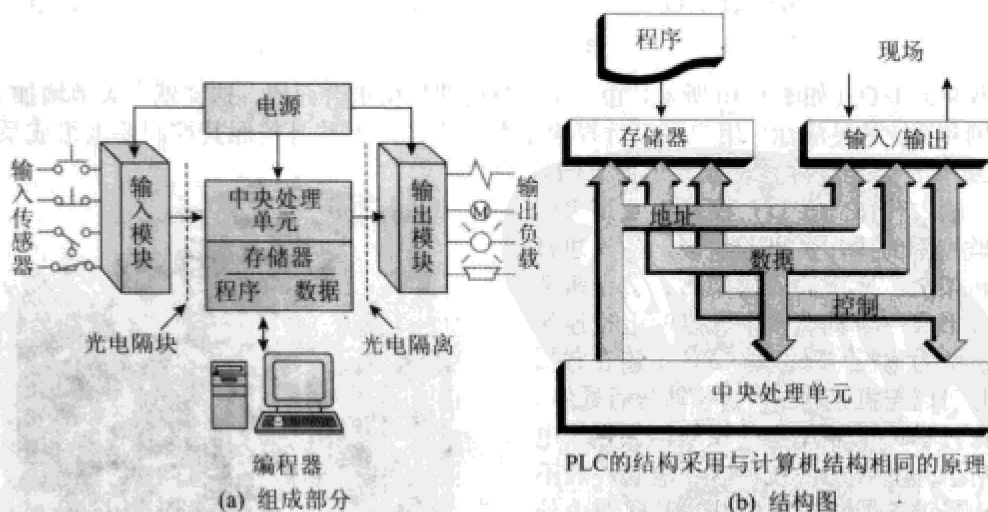
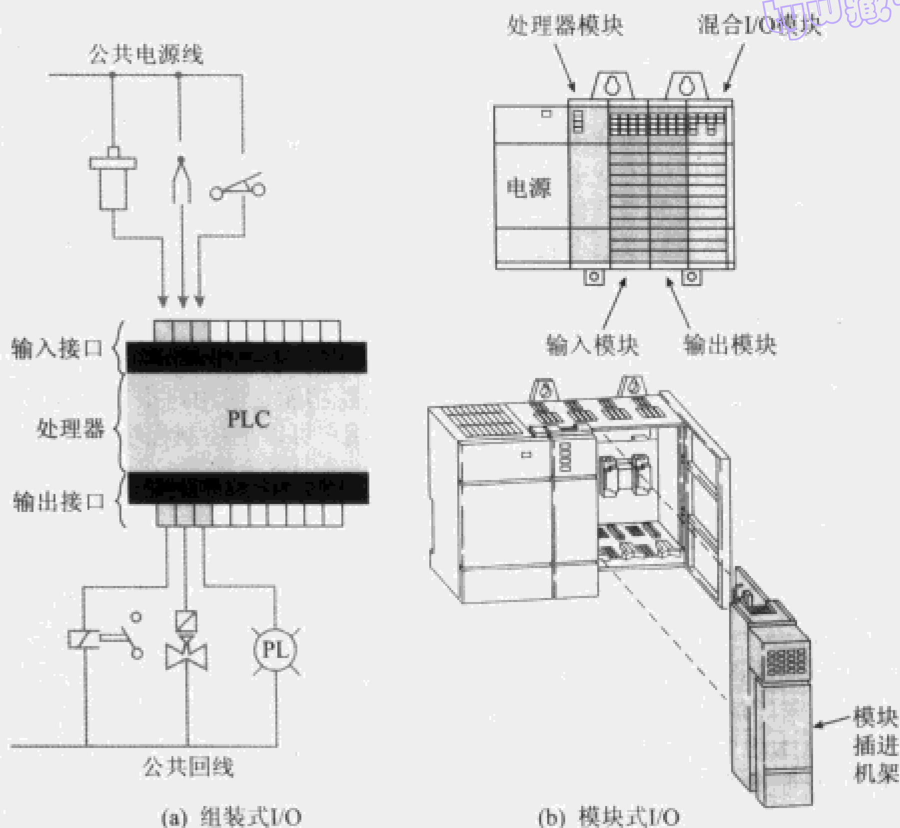


图 1-3 PLC 的组成部分和结构

在 PLC 内部有两种 I/O 连接方式：组装式和模块式。组装式 I/O（如图 1-4a 所示）是典型的小型 PLC，其中没有任何分离的和可移动的部件。处理器和 I/O 被组装在一起，并且 I/O 端子可以使用但不能改变。组装式 I/O 的主要优点是价格便宜。可使用的I/O点数是可变的，通常可以通过购买组装式 I/O 单元实现扩展。组装式 I/O 的一个缺点就是灵活性差，在使用时要受到组装的 I/O 所规定的数量和类型的限制。而且，对于有些模块，I/O 单元中任何一个部分出现故障，就只能换掉整个单元。



(a) 组装式I/O (b) 模块式I/O

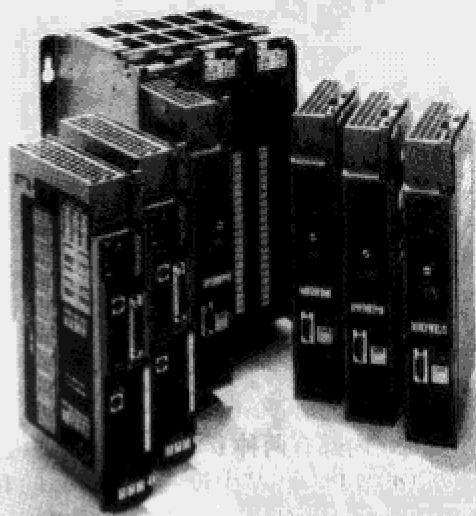
图1-4 I/O结构

模块式I/O（如图1-4b所示）由一个个分离的模块组合而成，这样就大大地增加了选择范围和系统的灵活性。用户可以选择模块及其生产厂家并且按照其控制要求组成系统。PLC基本的标准组件由机架、电源、CPU模块、输入/输出（I/O）模块以及用来编程和监控用的操作员界面组成。各模块都插在机架上。模块插入机架后，会在机架的背面形成一个具有多个触点的电气连接端子，称为背板。PLC的CPU也连在背板上并且可以与机架上的所有模块进行通信。

电源模块 为机架上的模块提供直流电源。对于大型PLC系统，这个电源一般不会向现场设备供电，现场设备的电源由外部交流电（ac）或直流电源（dc）提供。对于小型或微型PLC系统，这个电源也可以向现场设备供电。

7

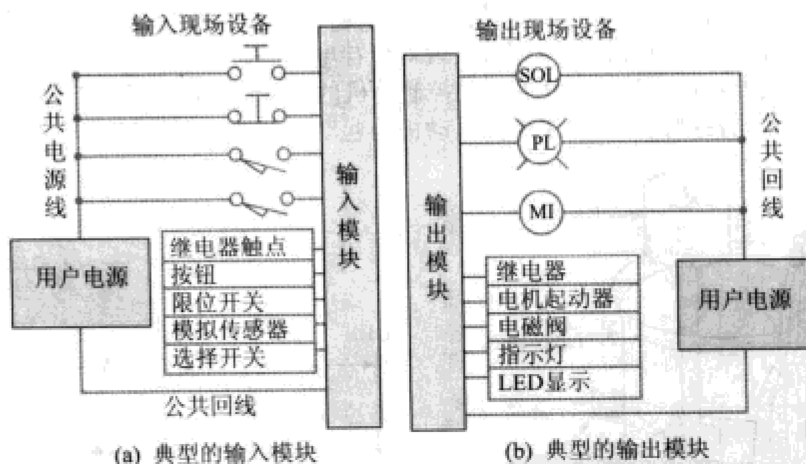
处理器（CPU）是PLC的"大脑"。典型的处理器（如图1-5所示）通常由一个微处理器和一个存储器组成。微处理器实现逻辑处理和控制各模块间通信的功能，存储器用于存放由微处理器完成的逻辑操


图1-5 典型的处理器模块（得到Allen-Bradley公司许可）

作的结果。存储器通常可采用 EPROM 或者 EEPROM 与 RAM 的组合。

PLC 具有 CPU，因此用户就可以用继电器梯形逻辑形式输入期望的电路到 CPU 中。处理器从各种传感器接收（读取）输入数据，执行存储在内存中的程序，然后发出适当的命令控制外部装置。CPU 工作时需要一个直流电源，提供处理器所需的低电平电压。根据生产厂家的不同，这个电源可以被安装在 CPU 内部，也可以是独立的电源模块。

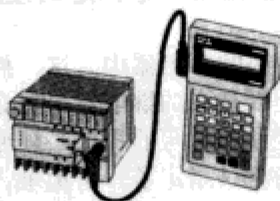
I/O 部分由输入模块和输出模块组成（如图 1-6 所示）。I/O 系统构成了现场设备与控制器连接的接口，其作用就是使从现场接收到的信号或者送到现场的信号达到处理器的要求。输入设备包括与输入模块接线端连接的按钮、限位开关、传感器、选择开关和指轮开关等；输出装置则是指与输出端连接的小型电机、电机起动器、电磁阀和指示灯等负载。内部元件与输入和输出的电气隔离采用光电隔离器，利用光耦合将两个电路连接在一起。我们把这些器件也归类于“现场”的输入和输出。术语“现场”（field 或 real world）用来区分那些实际的外部设备，这些设备的确存在并且需要与 CPU 内部担当继电器、计时器和计数器功能的内部用户程序有物理接线。



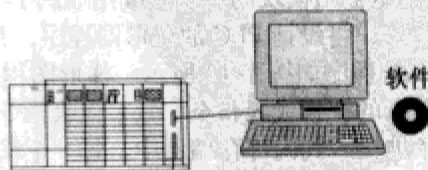
(a) 典型的输入模块　(b) 典型的输出模块

图 1-6

编程设备，或称编程终端，主要用于向存储器中输入用户程序。编写和输入用户程序所采用的梯形图程序语言是目前主要 PLC 生产厂家都采用的通用编程语言。梯形图程序语言采用图形符号，而非文字进行编程得到想要的结果。对于熟悉继电器控制逻辑的人员，采用梯形图语言编写 PLC 控制程序非常容易。手持编程器（如图 1-7a 所示）因为其价格便宜，使用方便，所以常用于小型 PLC 编程。当手持编程器与 PLC 相连的时候，手持编程器就可以用于输入和监控程序。紧凑的手持编程器常用于工厂现场的设备检修、修改程序和传送程序到多个 PLC 中。在一些小型手持编程器中，采用单独的布尔操作（AND，OR 和 NOT 函数）或组合形成逻辑语句的形式，把程序输入到 PLC。



(a) 带发光二极管显示的手持编程器



(b) 安装相应编程软件的个人计算机

图 1-7　编程设备

[9] 个人计算机（PC）是最常用的编程设备（如图1-7b所示）。所有主要品牌的PLC都有支持使用个人计算机作为编程设备的软件。用户使用这些软件可以编写、编辑、整理、存储和诊断梯形图程序并形成打印报告。计算机监控可以在屏幕上显示比手持编程器更多的指令，所以可简化程序的编译。计算机与PLC通信采用串行或者并行的数据通信链路。如果不使用编程单元，可以将编程器取下，这不会影响用户程序的执行。

附加可选的PLC配件常常还有：

- 人机接口装置，允许操作员输入数据或者监控数据。
- 远程I/O通信适配器，用于中央控制器与远程传感器和执行器的连接。
- 网络接口，使多个PLC互相连接或者与其他控制器构成分布式控制系统。

1.3 工作原理

我们以一个简单的过程控制问题为例来说明PLC的工作原理。图1-8所示为一液体搅拌系统，当桶内液体的温度和压力达到规定值时，混合电机开始自动搅拌液体，同时，还有一个独立的控制按钮用于电机的手动控制。当搅拌桶内的温度和压力达到设定值时，通过压力传感器和温度传感器开关的闭合实现对搅拌电机的起动控制。

这个控制问题可以用如图1-9所示的控制电机的继电器电路实现，当压力开关和温度开关接通或者手动控制按钮被按下时，电动机的起动线圈就被接通。

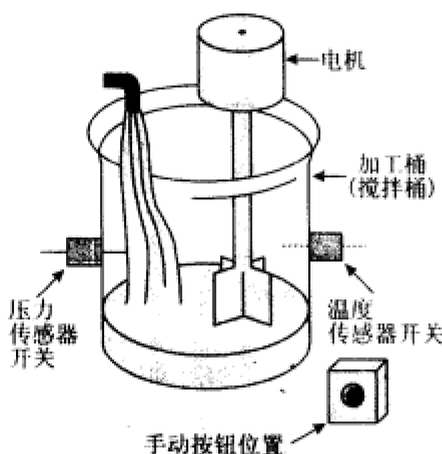


图1-8 混合过程控制问题

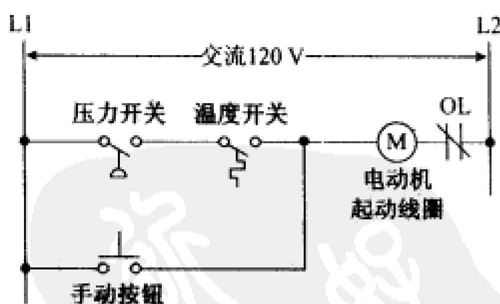


图1-9 过程控制继电器梯形图

[10] 现在，我们来看如何将PLC用于这样的系统中。首先，我们仍然需要压力开关、温度开关和按钮作为现场设备，直接与输入模块上标定的接线端相连。120V交流电输入模块的典型接线图如图1-10所示。

现场输出设备（电动机起动线圈）也同样需要与输出模块上标定的接线端相连。120V交流电输出模块的典型接线图如图1-11所示。

接下来，将设计PLC的梯形图程序，并输入到CPU内的存储器中。这一控制过程的典型梯形图程序如图1-12所示。梯形图程序采用了与硬件连线的继电器电路布局相似的格式。独立的符号代表指令，数字代表指令的地址。给控制器编写的程序必须通过编程器一条一条地输入到内部存储器中。每一个输入和输出设备都被指定了一个地址，这个地址让PLC可以区分它们的物理连接位置。但要注意不同的生产厂家PLC的I/O地址格式可能不相同。这些指令存放在存储器中的用户程序存储区。

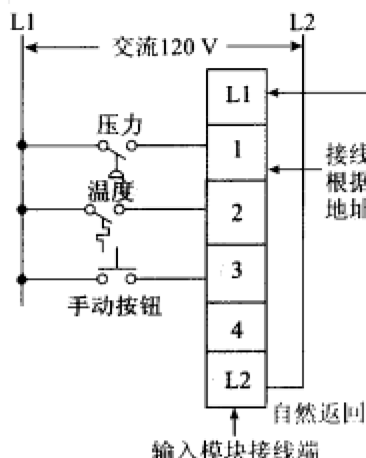


图 1-10 典型输入模块接线图

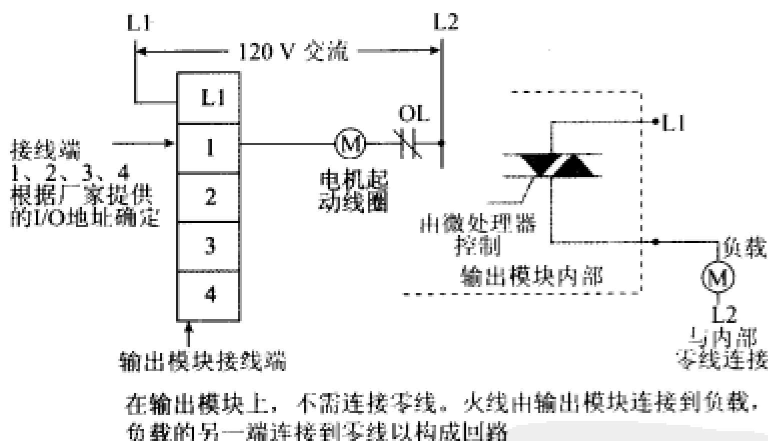


图 1-11 典型的输出模块接线

当 PLC 处于运行状态或工作周期时，就执行程序。在每一个工作周期，PLC 都完成检测输入设备的状态、执行用户程序并据此改变输出。每一个┤ ├代表一个常开触点，符号 -( )- 则代表线圈。对应的触点闭合时，将使线圈被激励。在图 1-12 所示的梯形图程序中，线圈 O/1 在触点 I/1 和 I/2 闭合或者触点 I/3 闭合时被激励。这两个条件中的任何一个满足时，就会提供一个从左到右包括线圈在内的环路。

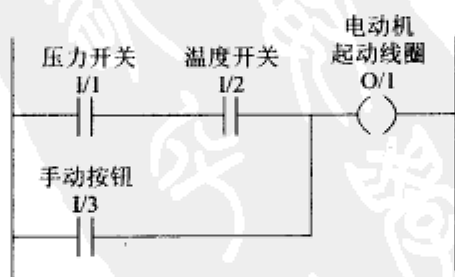


图 1-12 带地址的 PLC 过程控制梯形图

11

PLC 运行时的工作顺序可描述如下：首先，检测输入的状态并将其存储到控制器的内存中（一个闭合的触点状态被存储为逻辑 1，而断开的触点状态被存储为逻辑 0）。然后，根据存储的状态，执行梯形图程序，使内部触点闭合或者断开。如果这些触点为梯形图中电流提供一个从左到右的通路，则输出线圈的存储器位置将提供一个逻辑 1 的值，同时输出模块的触点将闭合。如果图中没有电流的通路，则输出线圈的存储器位置将提供一个逻辑 0 的值，同时输出模块的触点将断开。PLC 完成以上动作的周期被称为扫描周期。扫描周期是完成一次循环所需要的时间，它可

用于测量 PLC 的响应速度。一般来说，输出存储器的状态在扫描时不断更新，但是实际的输出更新则是在完成程序扫描之后的 I/O 扫描期间完成。

1.4　操作修改

12

正如前面所述，PLC 具有的重要特点之一就是修改程序方便。例如，如果液体混合操作的原始控制电路需要修改为如图 1-13 所示的继电器控制电路。即：当温度达到某个特定的设定值时，手动按钮在任何压力下都可以使起动线圈接通。

如果采用的是继电器控制系统，就需要按照图 1-13 重新接线，以完成其功能。然而如果使用 PLC 控制系统，就不需要重新接线。输入和输出仍然相同，需要改变的只是如图 1-14 所示的梯形图程序。



图 1-13　更改控制过程后的继电器电路

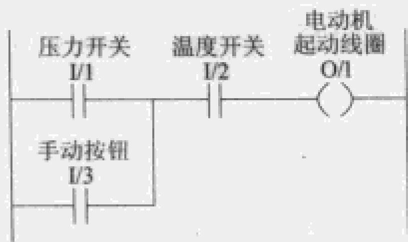


图 1-14　更改控制过程后的 PLC 梯形图程序

1.5　PLC 与计算机比较

PLC 的结构与一般的计算机大致相同。即使是个人计算机，如果提供适当的方法，使计算机能够接收例如按钮或者开关之类装置的输入信号，它也可以变成 PLC。计算机仍然需要程序来判断和处理输入信号的变化，然后决定是否驱动输出负载。

然而，PLC 与个人计算机之间还是有一些重要的区别。首先，和计算机不同的是，PLC 可工作在高温和潮湿的工业现场（如图 1-15 所示）。一个设计完善的 PLC 也不会受工业现场固有的电噪声干扰。

其次，PLC 的硬件和软件都便于在工业现场的工程技术人员使用。具体表现在 PLC 编程采用继电器梯形图逻辑或其他便于学习的编程语言，其编程语言被固化在 PLC 的永久存储器中。PLC 没有键盘、没有 CD 驱动器、没有监视器、也没有磁盘驱动器，但在 PLC 的内部有用于通信的端口和一组用于连接输入和输出装置的接线端。

13

计算机是复杂的计算机器，能够同时以不同的顺序执行几段程序或者任务；而大多数 PLC 都只能以一种顺序执行单一的程序，执行时从第一条指令开始依次连续执行到最后一条指令。

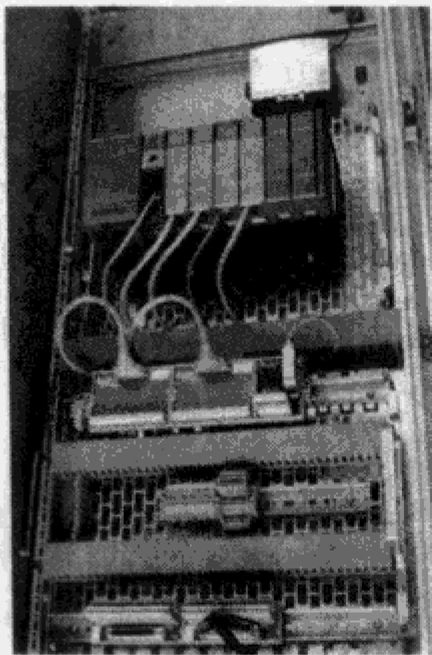


图 1-15　处于工业现场的 PLC（得到英国 Famic 公司许可）

也许 PLC 与计算机最显著的区别在于，PLC 的设计使得工厂电气人员在安装和维护时不需要高深

的专业计算机技术。因为大多数 PLC 都设计有错误指示并将错误信息显示在屏幕上，这使得 PLC 的故障诊断简化。与现场设备相连的模块接口实际上也是 PLC 的一部分，它们的连接和更换都很方便。

就像个人计算机已经改变了其他领域的做事方式一样，PC 也已经渗透到 PLC 工业控制领域。在 PC 上编写和运行软件已经改变了人们使用 PLC 的传统方式。通常，运行于 PC 机上的 PLC 软件有以下两种类型。

- PLC 软件允许用户编写程序和文件，它提供给用户一个编写 PLC 程序（梯形图或其他编程语言）的工具，并且可以根据需要给程序增加详细的注释。
- PLC 软件允许用户监控和控制操作过程（也称为人—机接口）。它使用户在 CRT 显示器上看到整个控制过程（或者用图形表示的过程），从而决定系统的运行、参数变化趋势和接收报警条件。

工业控制技术经历了气体动力学、电子继电器，到今天的用梯形图逻辑编程语言的固态 PLC 的发展过程。基于个人计算机的控制是传统的 PLC 技术发展的最新成果。尽管基于 PC 的控制已经开始多年，但花费这么多年生产的工具和系统对用户而言必然是有足够吸引力的。为了达到与一个完整的 PLC 相同的控制功能，计算机需要配置一些连接装置，以接收来自传感器和变换器的信息，并且驱动诸如指示灯、线圈、继电器和电机等负载。目前一些生产商已经开发了让个人计算机可以完成 PLC 功能的软件和相应的接口卡。这种系统我们相应的称其为*软逻辑控制器*。综上所述，基于个人计算机的控制系统有如下优点：

- 初始成本低；
- 对专用硬件和软件的需要少；
- 与其他系统可直接进行数据交换；
- 快速信息处理；
- 易于定制。

1.6 PLC 的规模与应用

一般来说，PLC 分为 5 类：超小型、微型、小型、中型和大型。用于 PLC 分类的标准包括：功能、I/O 点数、费用和物理尺寸。在所有这些因素中，I/O 点数是最重要的。例如：一个入门级的超小型 PLC 可以小到能放进衬衣口袋管理 16 个 I/O 点。微型 PLC 则可以连接多至 32 个 I/O 点。它们都具有大型 PLC 指令系统的 90% 以上的功能，适用于既需要采用专业控制，又不需要大型 PLC 那么多 I/O 点数的用户使用。超小型 PLC 和微型 PLC 在电梯控制、汽车清洗和搅拌机系统中使用极为方便。

另一方面，例如 Allen-Bradley SLC-500 系列小型 PLC，在一个机架（主机）上就能处理多达 960 个 I/O 点。小型和中型 PLC 还提供特殊的 I/O 模块，以提高控制系统的性能。为了提供一个在模块和处理器之间使用方便、统一的接口，这些模块的功能从模拟量运动控制到通信连接。大型 PLC-5 系列是 Allen-Bradley 公司功能最强大的 PLC，它们能处理几千个 I/O 点。这类 PLC 的应用范围几乎没有限制，它不仅可以用于独立的生产过程控制，也可以用于整个工厂的控制。

选择 PLC 的关键因素是准确地判断其用途，选择比实际需要大得多的 PLC 系统是不明智的。然而，选择的时候还是要首先预留出将来可能的扩展余地，以确保系统不仅能满足现在的要求，而且能够满足将来可能需要的应用。

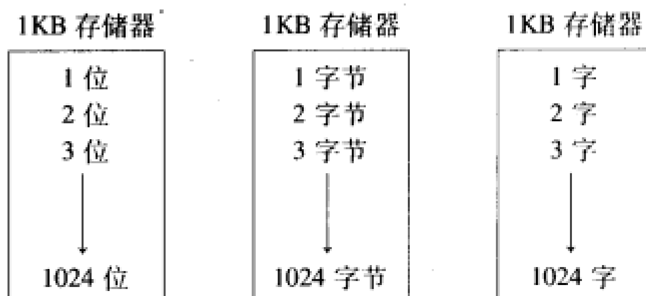
PLC 的应用主要分为三大类：单任务、多任务和控制管理。单任务控制的 PLC 的应用是指一个 PLC 控制一个生产过程，它是一个独立的单元，不能与其他的 PLC 或者计算机通信。很明显整个的控制过程的复杂度与系统的大小是选择 PLC 的重要因素。这种应

用场合可以使用一个大型的PLC，但是通常情况下都是用小型的PLC。

多任务的应用通常为中型PLC，此时PLC可同时控制几个生产过程。足够的I/O能力是这类PLC装置非常重要的因素。另外，如果这个PLC是一个大型生产过程的子系统，则它还必须与中央控制的PLC主机或者计算机构成网络，以完成相互间的数据通信。

PLC的控制管理就是一台PLC控制另外几台PLC。这类应用的要求是具有能与其他的PLC和计算机进行通信的大型PLC。作为控制管理的PLC是通过下载程序来管理其他的PLC，指挥它们工作的。通过适当的寻址方式，它必须能够随时与其中任何一个PLC进行通信。

PLC存储器的大小有多种，从最小的能适应小型系统的256字，到满足大型系统的2M（Mega）B。存储器的容量通常用KB值表示：1KB、6KB、12KB等。K是计量单位kilo的缩写，通常表示1000个单元。然而，我们在说到PLC或者计算机存储器的大小时，1K代表的是1024，因为这个计量单位取决于二进制系统（$2^{10}=1024$）。根据存储器的类型，1KB可以表示1024 bit，也可以表示1024字节或者1024字。



尽管我们通常用“字”来计量PLC的存储容量，但是在对存储容量进行精确比较时，还是需要知道每个字的位数。例如，一个具有6KB字存储容量的PLC，如果采用8 bit字，则存储器有49 152 bit（$8\times6\times1024=49\ 152$）；如果采用16 bit字，则存储器有196 608 bit（$32\times6\times1024=196\ 608$）。所以，存储器的需要量取决于它的应用。影响一个具体的PLC安装所需要的存储器容量的因素包括：

- I/O点的使用量；
- 控制程序的大小；
- 数字采集的需求；
- 需要的管理功能；
- 扩展功能。

一个实际的PLC指令系统包含各种不同类型的指令。具体来说，小型机的指令的条数只有15条，而功能强大的大型PLC则有100条指令（如表1-1所示）。

自从PLC产生以来，它已经成功地应用到了包括轧钢机、造纸纸浆厂、化工厂和汽车生产厂、电厂等几乎所有工业领域。PLC完成了从简单机器的重复性开关控制到复杂的生产过程控制等众多不同类型的控制任务。

表1-1 典型的PLC指令

指令	操作
XIC（ON检测）	检测ON条件的位
XIO（OFF检测）	检测OFF条件的位
OTE（输出激活）	接通一个位（非保持型）

（续）

指　令	操　作
OTL（输出闭合）	接通并保持一个位（保持型）
OTU（输出断开）	断开并保持一个位（保持型）
TOF（断开延时定时器）	在定时器线圈断开时，使输出在事先设置的时间间隔后接通或者断开
TON（接通延时定时器）	在定时器线圈通电时，使输出在事先设置的时间间隔后接通或者断开
CTD（减计数）	用软件计数器对给定值减计数
CTU（加计数）	用软件计数器对给定值加计数

16

思考题

1. PLC 的定义是什么？
2. 列举出 7 个 PLC 与传统继电器控制系统相比的突出优点。
3. 列举出 4 个继电器开关任务之外的 PLC 能够完成的任务。
4. 叙述 I/O 与 PLC 连接的两种方法。
5. 描述将 I/O 模块与模块化 PLC 的 CPU 连接的方法。
6. 描述 PLC 的下列部件的主要功能：
   a. CPU 模块　b. I/O 模块　c. 编程器　d. 电源模块
7. 举出两种常用的 PLC 编程设备。
8. 参考图 1-9 所示单元过程控制的继电器电路，回答下列问题：
   a. 压力开关什么时候闭合？　b. 温度开关什么时候闭合？　c. 压力开关和温度开关是以什么形式相互连接在一起的？　d. 描述使电机起动线圈得电的两个条件。　e. 当下面的开关处于断开状态，在每一个开关上面的电压降大约是多少？
   （1）压力开关　（2）温度开关　（3）手动按钮
9. 参考图 1-12 所示的单元过程控制 PLC 梯形图，回答下列问题：
   a. 各个符号代表什么？　b. 数字代表什么？　c. I/2 表示什么现场设备？　d. O/1 表示什么现场设备？　e. 使程序变为从左到右连续通路的两个条件是什么？　f. 描述 PLC 程序在一个扫描周期的工作顺序。

17

10. 比较继电器系统和 PLC 系统改变过程控制操作时采取的方法。
11. 从如下方面将 PLC 与通用计算机进行比较：
    a. 工作环境　b. 编程方法　c. 程序的执行　d. 维护
12. 描述两种能在个人计算机上编写和运行用来与 PLC 相连接的软件。
13. a. 根据如下的描述，确定 PLC 的规模类型：
    （1）上千个 I/O 点　（2）32 个 I/O 点　（3）小到可以放进衬衣口袋
    b. 选择 PLC 规模的两个最重要的因素是什么？
14. 比较用于单任务、多任务和控制管理类型的 PLC 的应用。
15. 列出影响实际安装的 PLC 存储器容量的 5 个因素。
16. PLC 的指令系统是什么？
17. 什么是软逻辑控制器？
18. 为什么目前所设计应用的大多数 PLC 控制系统都被认为是专用的？
19. 指出 PLC 与个人计算机在物理硬件上的区别。
20. 最流行的 PLC 编程语言是什么？为什么？

21. PLC工作在实时状态是什么意思？

18

22. 一个具有8K字容量的采用16位字的PLC，它的容量用位表示是多少？

习题

1. 用两个单极开关，编写一个满足以下要求的程序：当开关A和开关B同时闭合时，输出接通。
2. 用两个单极开关，编写一个满足以下要求的程序：当开关A或开关B闭合时，输出都接通。
3. 用四个常开按钮（A，B，C，D），编写一段程序，使之满足：按下按钮A和B或者按下按钮C和D，灯都发光。
4. 编写实现如图1-16所示的继电器梯形图的程序。

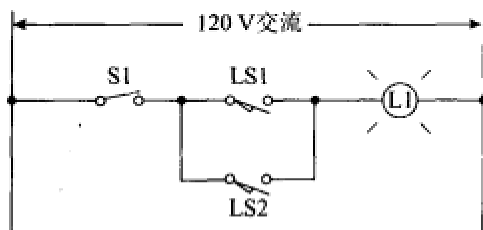
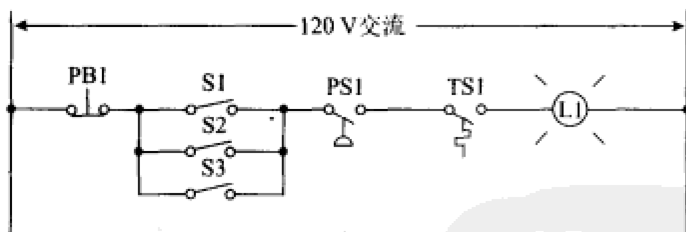


图 1-16

5. 编写实现如图1-17所示的继电器梯形图的程序。



19

图 1-17

第 2 章　PLC 的硬件组成

学习目标：

- ❑ 列出 PLC 系统中的硬件元件，并描述其功能。
- ❑ 描述基本电路、离散和模拟模块的运用，并且描述常用输入/输出模块和 CPU 的规格。
- ❑ 解释 I/O 端的编址。
- ❑ 描述 PLC 存储设备的常用类型和型号。
- ❑ 列出不同种类的 PLC 外围辅助设备，并对其进行描述。

本章将详细介绍 PLC 的硬件以及构成 PLC 控制系统的模块。对不同的 PLC 元件和一般连接路径进行说明。将对 CPU 和不同类型的存储硬件设备进行讨论，同时也对输入/输出部分包括离散类型模块和模拟类型模块之间的区别进行说明。 20

2.1　I/O 部分

对于 PLC 的“大脑”来说，输入模块和输出模块是 PLC 的“眼”、“耳”和“舌”。I/O 部分包括如图 2-1 所示的 I/O 机架和 I/O 模块。输入模块将从机器或加工设备接收到信号，转换为 CPU 可以使用的信号。输出模块将 CPU 输出的信号转换为用于控制机器或流程的外部信号。通常，PLC 都具有几个 I/O 模块的扩展空间，允许根据一定的需要，选择恰当的模块来扩展 I/O 模块。PLC 上的槽可以固定任何类型的 I/O模块。

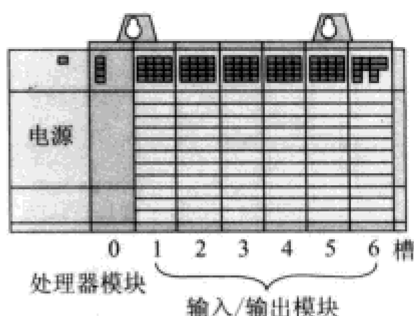


图 2-1　I/O 部分

I/O 模块在现场和 CPU 之间提供了一个的硬接线的接口。输入接口允许将与流程相关的状态信息传送到 CPU，且允许通过输出接口将 CPU 的操作信号传送到由它控制的设备中。

机座是用于放置设备的物理硬件，可以用来安放 I/O 模块、处理器模块和电源。机座按照它所包含槽的数目可分为不同的尺寸。通常情况下，机架具有 4 个、8 个、12 个或 16 个槽。

逻辑机架是一个由 128 个输入点和 128 个输出点构成的可编址单元。机架在输入映像表存储区和输出映像表存储区中各占有 8 个字的空间。在输入映像表存储区中的一个字和与其相关的位于输入映像表存储区中的字被称为一个 I/O 组。在具有 128 个离散的 I/O 点时，一个机架最多能够包含 8 个 I/O 组（编号从 0 到 7）（如图 2-2 所示）。在一个机座上可以安装不止一个逻辑机架，在一个逻辑机架上也可以连接不止一个机座。

PLC 系统优点之一是能够将 I/O 模块安置于靠近现场设备的地方，使得接线量最小。当逻辑机架被安置在远离 CPU 模块时，此机架（如图 2-3 所示）被称为远程机架。为了和 CPU 进行通信，远程机架利用了一个特定的通信网络。每个远程机架都需要一个独立的通信站号，使之和其他机架区分开来。远程机架通过一个通信模块和现场机架进行连接。通过电缆将各机架连接起来。如果在 CPU 和 I/O 机架间使用了光纤电缆，就能够在 32 km 以外的地方对 I/O 点进行操作而不产生压降。同轴电缆允许远程 I/O 机架被安装在 21~22

离 CPU3.2 km 的地方。光缆不会携带由邻近的高压电流或工业环境设备所产生的噪声。相比之下同轴电缆对噪声更敏感。

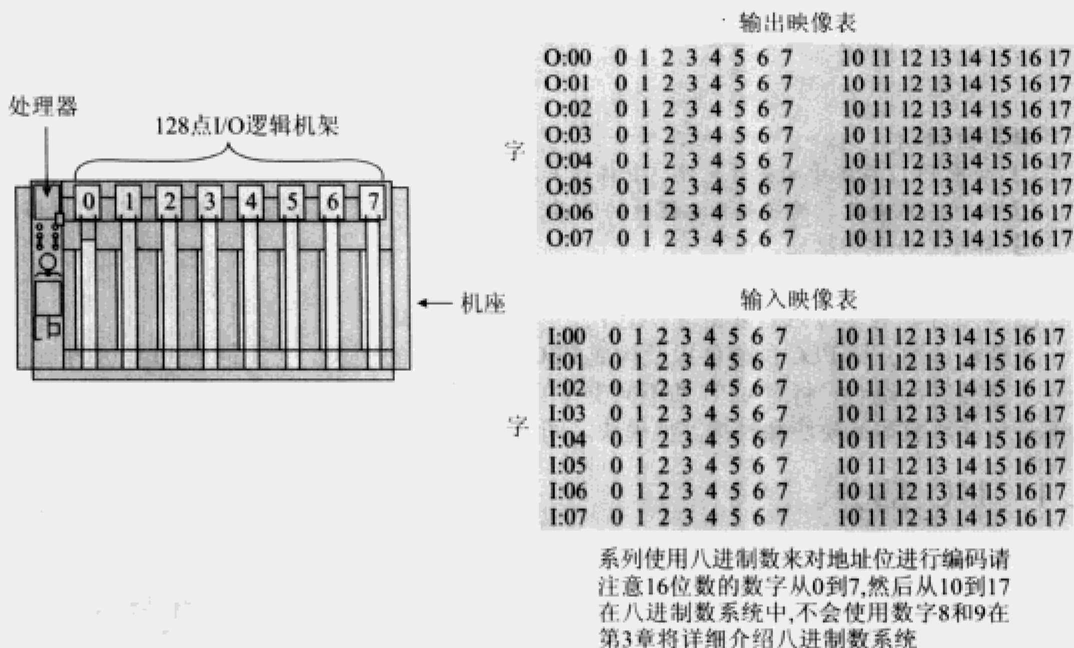


图 2-2　逻辑机架

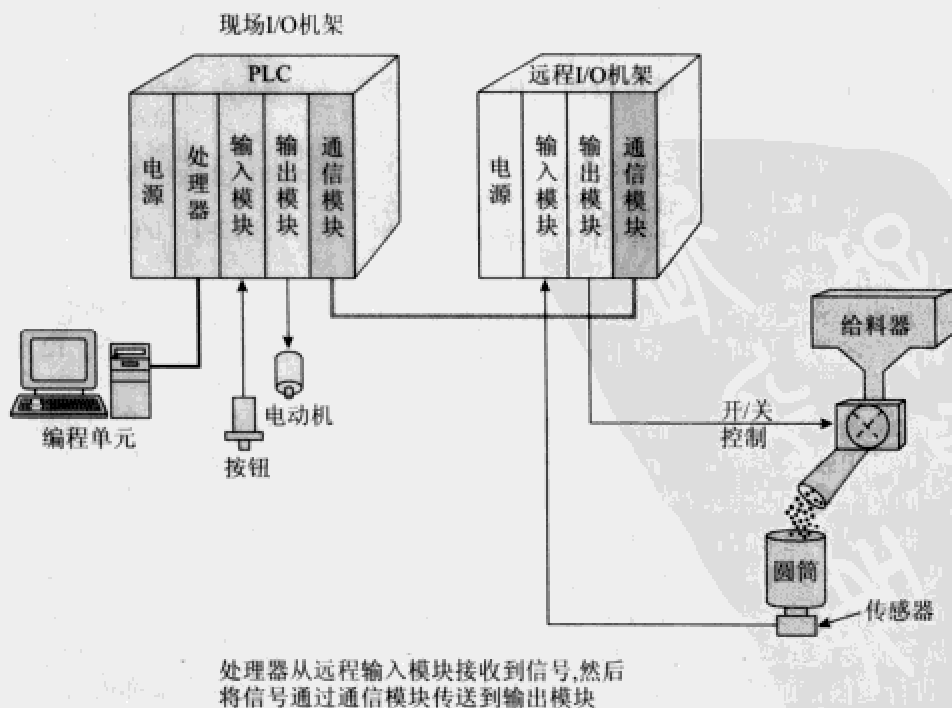
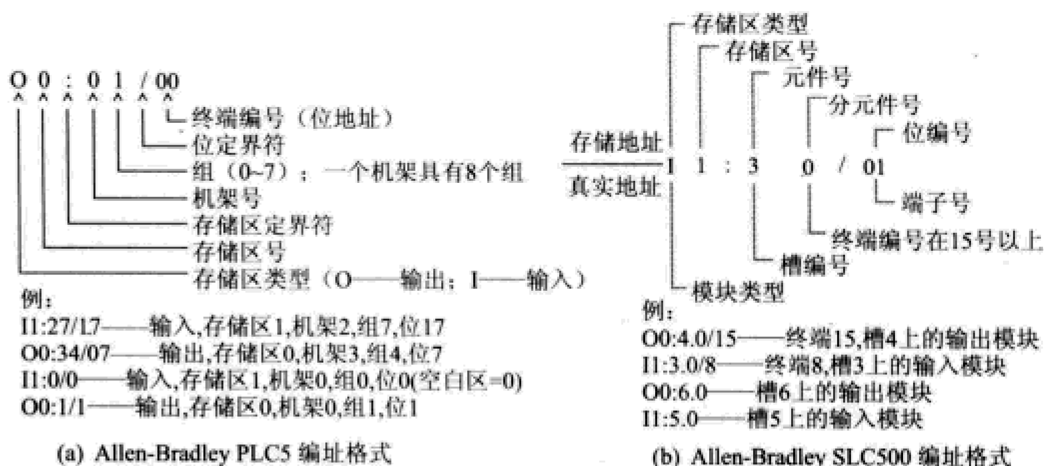
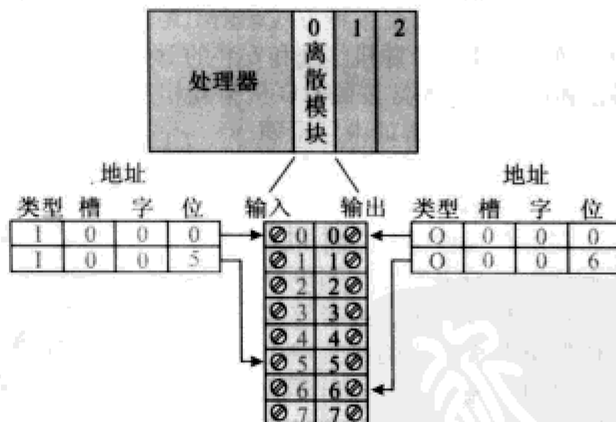


图 2-3　远程 I/O 机架

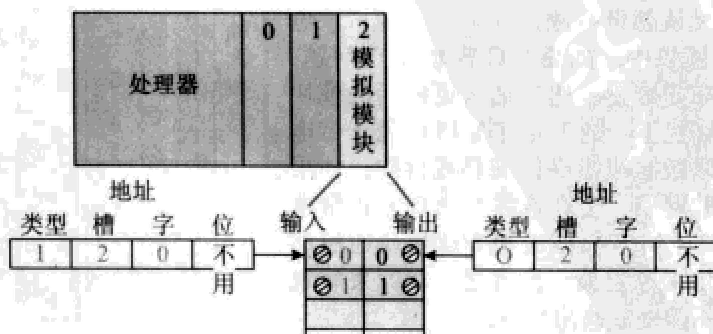
模块在机架中的位置，以及和输入或输出设备相连接的模块接线端数目，将决定设备的编址（如图 2-4 所示）。每个输入和输出设备都必须具有特定的地址。处理器将根据这些地址来辨识被监控的设备的位置。另外，I/O 模块在现场的接线方式也有很多种。现场设备和 I/O 模块的接线使更换模块时的拆线和再接线变得更容易。各个模块同样具有显示每个I/O电路的 ON 或 OFF 状态的指示灯，并且大多数输出模块还具有保险丝熔断指示器。



(a) Allen-Bradley PLC5 编址格式　　(b) Allen-Bradley SLC500 编址格式



(c) 离散I/O模块编址



(d) 模拟I/O模块编址。通常地址位没有使用；然而，如果有必要，由数字量代表的模拟值也可用编程器进行编址

图 2-4 I/O 模块编址



(e) 符号地址是真实的名称或代码，编程器可以用逻辑地址来替换它们，这是因为它们和实际运用相关。在此例中，编址符号为LS_3和pump_14，而相应的实际地址为I:3/3和O:4/14。Allen-Bradley逻辑控制器使用一种符号编址模式。每个位由一个用于辨识控制其中存储地址的标签名称来表示

图 2-4 （续）

通常情况下，基本的编址因素包括：

- 类型

23 类型决定了输入端或输出端是否正在被编址。

- 槽

槽的编号表示 I/O 模块安放的物理地址。当使用扩展机架时，可以将机架编号和槽的编号进行结合。

- 字和位

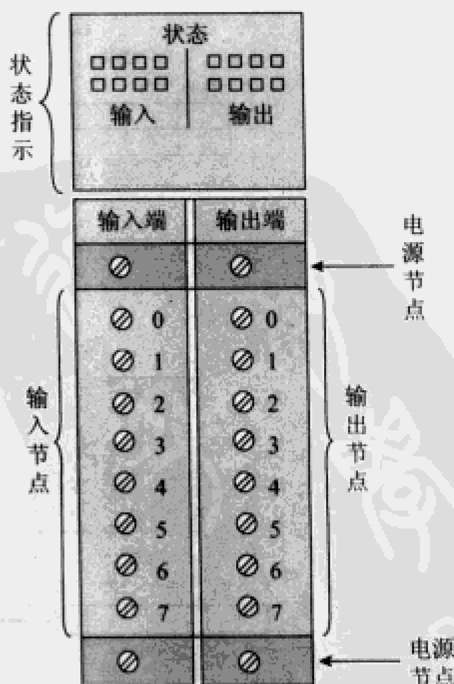
字和位用于辨识特殊 I/O 模块的实际终端连接。离散模块通常只使用一个字，且接线与构成字的不同位相关。PLC 的设计决定了系统是否能够采用可变的编址方式还是采用固定的编址方式。PLC 系统设计师利用可变编址方式编创出逻辑控制软件，不需要按照一定顺序进行 I/O 地址分配，而是产生一个随机编址和安装的 I/O 系统。

24 在柔性系统中，各槽和接点地址通常按照 I/O 机架互相连接的顺序来决定。在一些小型 PLC 中系统有一个机架，因此 I/O 编址由厂家确定。实际的地址标签根据厂家的不同而有很大不同。

标准的 I/O 模块由印制电路板和类似于图 2-5 所示的接线端所构成。印制电路板就是一个将 PLC 和输入或输出设备联系起来的接口电路。印制电路板可以插入槽或 I/O 机架中的接线端中，或者直接安置在处理器中。位于印制电路板前沿的接线端用于现场接线。模块中还包含每个输入点和输出点的接线端、每个输入端和输出端的状态指示灯和向输入端和输出端供电的电源节点。

大多数 I/O 模块都带有插入式扁平电缆，该电缆被插入到实际模块中。如果 I/O 模块出现问题，可以将整个扁平电缆取出，然后插入更换后的模块中。除了在特定情况下，绝不允许在 PLC 处于带电状态下对 I/O 模块或接线板进行安装或拆除。如果模块插入到错误的槽中，会在连线上出现非正常电压，导致模块损坏。大多数面板和 I/O 模块通过加密来防止模块被错误地安装。换言之，在原来放置

25 输入模块的槽中不能安放输出模块。



接线端、状态指示器和电源节点的布置随厂家的不同而变化。注意I/O模块既有输入端也有输出端

图 2-5 典型 I/O 模块组合

输入和输出模块可以安置在机架上的任何部位，但是为了接线方便，通常将它们组合起来放在一起。I/O 模块可以具有 8 个、16 个或 32 个输入/输出点。点数指的是可用的输入端或输出端的数目。标准 I/O

模块具有 8 个输入端或输出端。高密度模块可以有多达 32 个输入端或输出端。高密度模块的优点是能够在一个槽中安置 32 个输入端或输出端，因此节省了大量空间。唯一的不足是高密度输出模块每个输出端不能处理大的电流。32 点的插件通常拥有至少 4 个公共节点。

2.2 离散 I/O 模块

大多数常用的 I/O 模块都是离散型的。其输入模块常与 ON/OFF 性质的输入设备相连，如选择开关、按钮和限位开关等。输出模块则局限于指示灯、小型电动机、螺线管和电机启动器等只执行简单 ON/OFF 操作的设备。离散 I/O 模块的分类包含位定向的输入端和输出端。在这种类型的输入端和输出端中，每个位都表示一个元件的完整的信息，并且提供一些外部触点的状态，或电路的来电、失电状态。

每个离散 I/O 模块都是由现场的电压源供电。由于电压类型或幅值的不同，I/O 模块分为不同的交流或直流等级，如表 2-1 所示。插在机架上的模块从机架后端获得正常运行所需要的电压和电流。电源为位于 I/O 模块电路板上的有源或无源电子元件供电。根据负载的需要，相对较大的电流，由用户提供的电源提供。模块的电源等级可以是 3A、4A、12A 或 16A，这取决于所使用的模块的类型和数量。

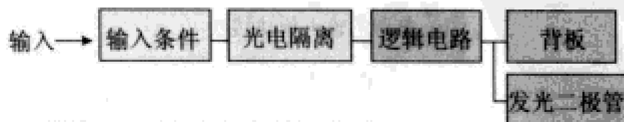
表 2-1 离散 I/O 接口模块的常用等级

输入接口	输出接口
12V 交流/直流/24V 交流/直流	12V – 48V 交流
48V 交流/直流	120V 交流
120V 交流/直流	230V 交流
230V 交流/直流	120V 直流
5V 直流（TTL 电平）	230V 直流
	5V 直流（TTL 电平）
	24V 直流

图 2-6 所示为一个典型的交流离散输入模块的一个输入端的等效框图。输入电路由两个基本部分构成：电源部分和逻辑部分。电源部分和逻辑部分通常由一个电隔离电路将它们分开。 26



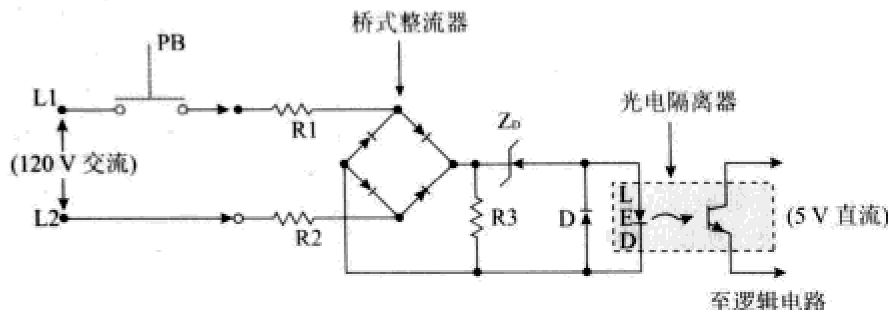
(a) 离散输入模块的框图



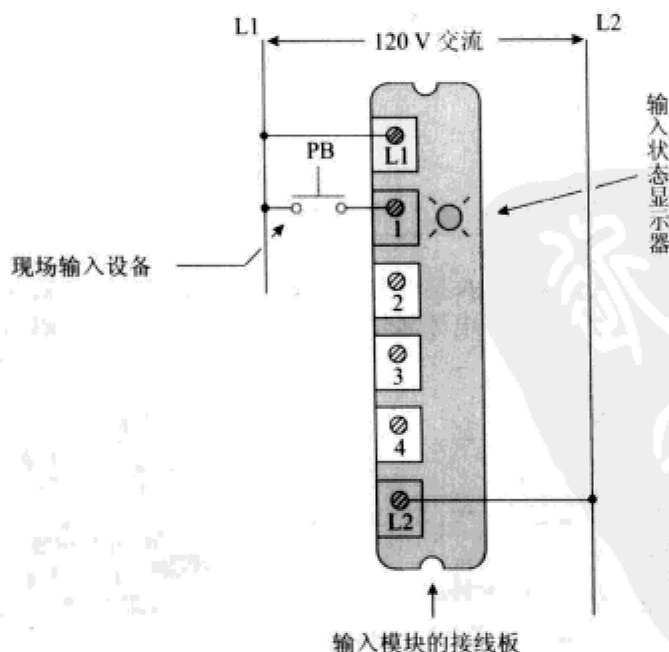
(b) 输入电路按照以下方式来反映输入信号：
- 输入滤波器滤除导致触点跳动或电子干扰的有害信号
- 光电隔离器通过将逻辑电路和输入信号隔离开，从而保护输入电路和背板电路
- 逻辑电路对信号进行处理
- 输入LED的开和关，表明了相应输入设备的状态

图 2-6 交流离散输入模块

图 2-7a 和图 2-7b 所示为典型的交流输入模块的输入端简化电路和接线图。当按钮闭合时，120V 交流电通过电阻 R1 和 R2 加在桥式整流器的两端。整流器输出一个降低的直流电压，此电压加在光电隔离器的发光二极管上。齐纳二极管的电压等级设置为能够被检测到的最低电压等级。当发光二极管的光照射到光敏晶体管时，晶体管变为导通，且按钮状态通过逻辑电平或低电平的直流电压传送到处理器。光电隔离器不仅将高交流输入电压和逻辑电路分开，而且防止由于线电压发生瞬变而损坏处理器。光电隔离也可以减少工业环境中电子噪声的影响，这些噪声通常会造成处理器的不正常工作。通过使用脉冲转换器也可以实现耦合和隔离。



(a) 交流输入模块的简化电路



(b) 典型的输入模块接线

图 2-7

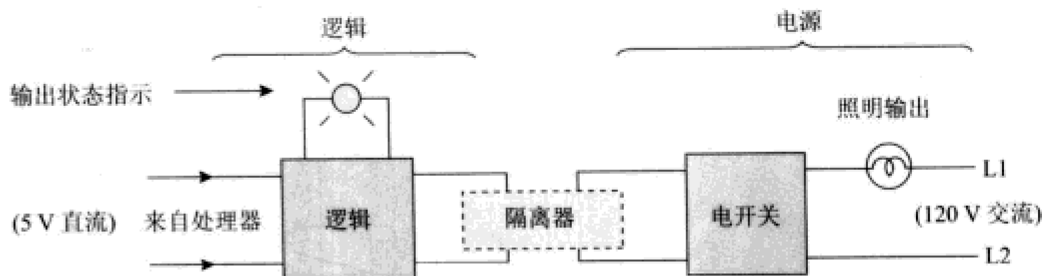
输入模块通常用发光二极管（LED）来监控输入端。每个输入端都有一个 LED。如果输入端为 ON，则 LED 为 ON。在故障检修时，模块上的 LED 非常有用。

在 PLC 控制系统中，输入模块需执行 4 个任务：

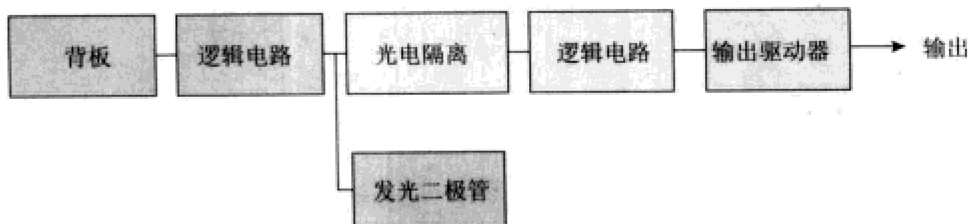
- 检测从机器上的传感器接收到的信号。

❑ 将输入信号转换为适用于特定 PLC 电压值的信号。
❑ 将 PLC 的有用信号从输入电压或电流信号的波动中分离出来。
❑ 将信号传送到处理器，使之能显示是哪一个传感器所发出的信号。

图 2-8 所示为一典型离散输出模块的框图。和输入模块一样，输出模块也由两个基本部分组成：电源部分和逻辑部分，这两部分通过隔离电路进行耦合。输出接口可以被认为是一个简单的电子开关，用于控制输出设备。 [28]



(a) 离散输出模块的框图

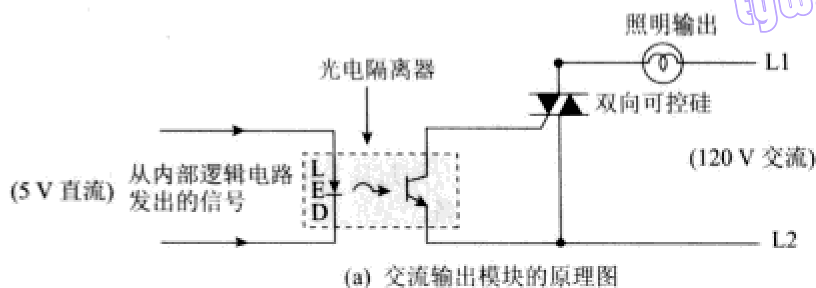


(b) 输出电路按照以下方式对输出信号进行控制：
- 逻辑电路决定输出状态
- LED显示输出信号的状态
- 光电隔离将输出逻辑电路和背板电路同现场信号分开
- 输出驱动器使得相应的输出端ON或OFF

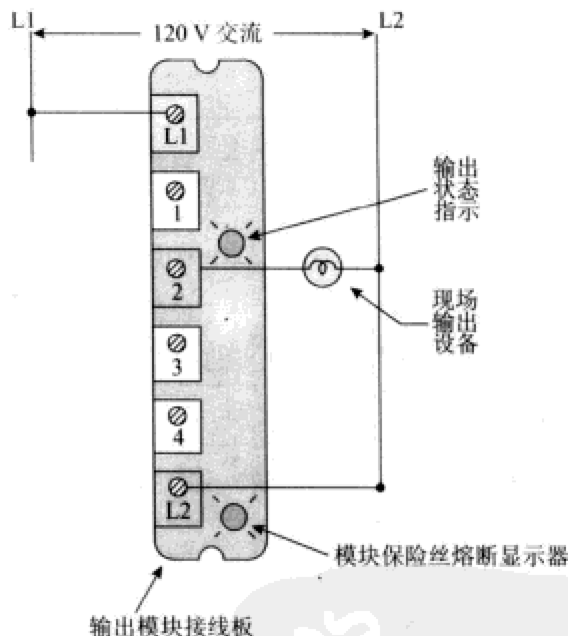
图 2-8 交流离散输出模块

图 2-9a 所示为一典型的交流输出模块的一路输出简化电路和接线图。正常的工作过程是：CPU 根据逻辑程序来设置输出状态。当 CPU 发出输出信号时，隔离器的 LED 两端将加有电压。LED 发光，使得光敏晶体管导通。从而使双向可控硅导通，使灯发光。由于双向可控硅能够双向导通，因此向灯输出的是方向可变的电流。双向可控硅没有 ON 和 OFF 状态，只有相对的低阻抗和高阻抗两种状态。在它的 OFF 状态（高阻抗），仍然有毫安级的漏电流流过双向可控硅。和输入电路一样，输出接口也提供了显示各输出端状态的 LED。在输出模块中通常还需要配有熔断器，它们安装在每个输出电路基线上，因此能够保护每个电路。一些模块还对熔断器状态提供指示器显示（图 2-9b）。 [29]

单个的交流输出受双向可控硅的大小所限，输出电流通常为 1A 或 2A。任何模块的最大负载电流也是有规定的。为了保护输出模块电路，不应该超出特定的电流范围。为了控制更大的负载，如大电动机，需要将标准控制继电器和输出模块相连。然后利用继电器的触点来控制大负载或电动机起动器，如图 2-10 所示。当按此方法使用控制继电器时，通常称为插入继电器。



(a) 交流输出模块的原理图



(b) 典型输出模块的接线

图 2-9

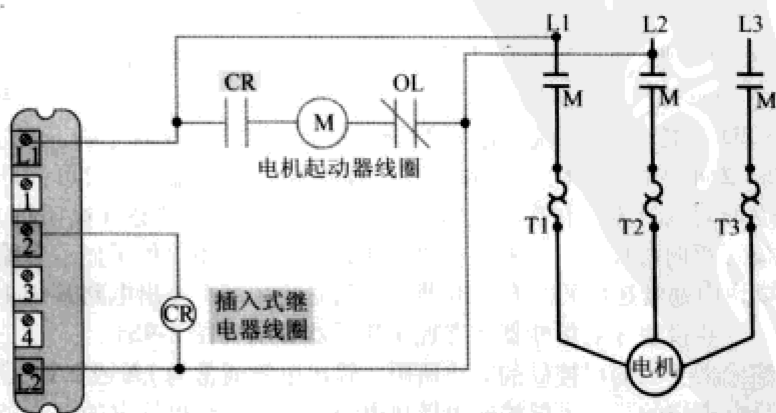


图 2-10 插入式继电器接线

离散输出模块用来控制实际输出设备的开和闭。这些模块可以用于控制任何具有两种状态的设备，并且适用于交流和直流两种形式，以及不同的电压和电流范围。输出模块中

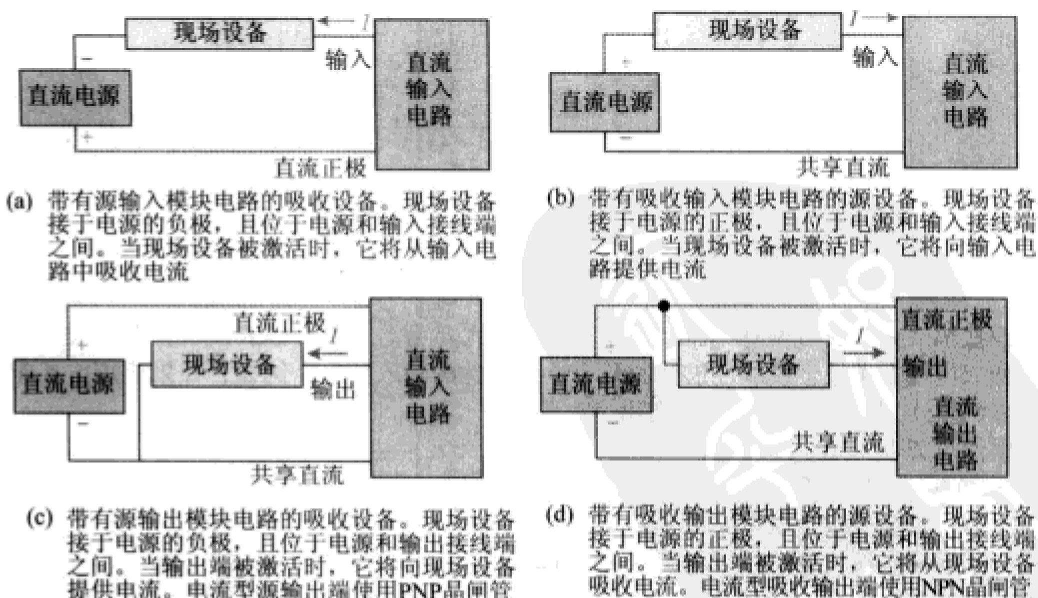
可使用晶体三极管、双向可控硅或继电器输出。双向可控硅输出只用于控制交流型设备，而晶体三极管输出只用于直流型设备的控制，继电器输出既可以用于交流型设备也可用于直流型设备，但是它的开关速度和固态输出相比显得比较慢。Allen-Bradley 模块利用彩色编码来辨识模块的类型，如下所示：

颜色	I/O 类型
红色	交流型 I/O
蓝色	直流型 I/O
橙色	继电器输出
绿色	特殊模块

直流现场设备的设计通常根据设备内部电路的要求，使用特殊的吸收电路或源电路。术语吸收和源是用于描述控制系统中现场输入和输出设备与电源之间的电流信号的流通关系（如图 2-11 所示）。这些直流输入和输出电路通常和现场设备搭配使用，设备内部固态电路的工作需要提供直流电压信号。和电源正极（+V）相连的现场设备为源设备。和电源负极（共享的直流信号）相连的现场设备为吸收设备。为了保持现场设备和 PLC 系统的电气一致性，这个定义扩展到离散直流 I/O 模块中的 I/O 电路中。

30

- ❑ 源 I/O 电路向现场设备提供（电源）电流。
- ❑ 吸收 I/O 电路从现场设备接收（吸收）电流。



(a) 带有源输入模块电路的吸收设备。现场设备接于电源的负极，且位于电源和输入接线端之间。当现场设备被激活时，它将从输入电路中吸收电流

(b) 带有吸收输入模块电路的源设备。现场设备接于电源的正极，且位于电源和输入接线端之间。当现场设备被激活时，它将向输入电路提供电流

(c) 带有源输出模块电路的吸收设备。现场设备接于电源的负极，且位于电源和输出接线端之间。当输出端被激活时，它将向现场设备提供电流。电流型源输出端使用PNP晶闸管

(d) 带有吸收输出模块电路的源设备。现场设备接于电源的正极，且位于电源和输出接线端之间。当输出端被激活时，它将从现场设备吸收电流。电流型吸收输出端使用NPN晶闸管

图 2-11 吸收和源

2.3 模拟量输入/输出（I/O）模块

早期的 PLC 由于只限于离散 I/O 接口界面，它只能连接具有开/关两种状态的设备。这种局限性使得 PLC 只适用于部分流程控制。现在，PLC 既有 I/O 离散接口又有模拟 I/O 接口，这使得 PLC 可以运用到任何类型的控制流程中去。

离散设备的输入端和输出端只具有两种状态：开和闭。模拟设备的输入端和输出端具有无数个状态。这些设备不仅具有开和闭两种状态，而且具有刚刚打开、几乎全开、几乎

关闭等状态。它们将向 PLC 发出或从 PLC 接收复杂的信号。

模拟量输入接口模块包括必要的能够接收现场设备所发出的模拟电压或电流信号的电路。通过模/数（A/D）转换电路，可以将输入信号从模拟量转换为数字量。模拟信号成比例的转换值，被处理器转换成 12 位二进制数或 3 位二进制编码的十进制（BCD）数来使用。传感设备的模拟量输入信号包括温度、光、速度、压力和位置转换等。图 2-12 所示为一个同热电偶相连的典型模拟量输入模块。热电偶产生的可变的毫伏级直流电压与被测温度成正比。将此电压放大并且通过模拟量输入模块进行数字化，然后以程序指令的形式送入处理器。由于输入信号的电压很低，电路的接线需要使用屏蔽线，以减少外部电子噪声。因为噪声信号可能引起瞬时的误操作，从而导致机器产生危险的或无法预测的结果。

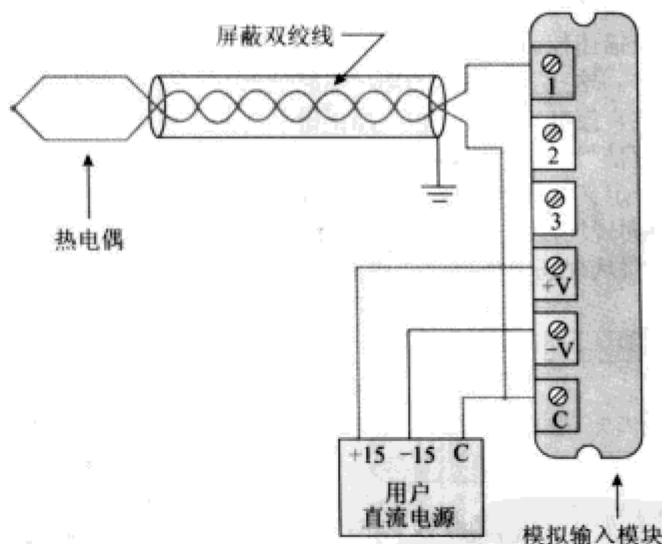


图 2-12 典型热电偶和模拟输入相连模块

模拟输入模块具有两种基本类型：电流型和电压型。电压型输入模块分两种类型：单极性和双极性。单极性模块能够接收只有一个极性的输入信号。例如，如果应用中要求插件能够测量 0 到 10V 的电压值，则用单极型。双极型插件可以接收正极输入和负极输入。例如，如果在应用中产生一个介于 -10V 和 +10V 之间的电压值，由于被测电压可正可负，则要用双极型。电流型输入模块通常用于测量 4mA 到 20mA 之间的电流。

模拟量输出接口模块能够从处理器接收数字信号，然后将其转换为一定比例的电压或电流来控制现场模拟量设备。通过数/模（D/A）转换电路可以将数字数据转换为模拟量的形式。模拟输出设备包括小型电动机、操作阀和模拟计量器。

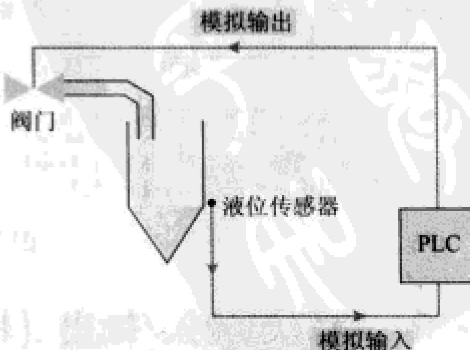


图 2-13 典型模拟 I/O 控制系统

图 2-13 说明了一个实际 PLC 控制系统中模拟 I/O 模块的使用情况。在应用中，通过调整阀门的开度，PLC 控制流入容器液体的流量。阀门的初始开度为 100%。随着容器中的液位达到预置点，PLC 将改变输出信号，从而调整阀门维持在一个设置点。

2.4 特殊 I/O 模块

特殊输入/输出模块用于满足某些特殊的需要，包括以下几种

- **高速计数模块**

计数器的速度超过 PLC 梯形图程序的权能，就需要用高速计数器模块提供的接口来完成。高速计数器模块用于对高速工作的传感器、编码器和开关所发出的脉冲信号进行计数。它们能够独立于处理器而进行计数。其典型计数的频率为 0 到 75kHz，这意味着模块每秒最多可以计数 75 000 个脉冲信号。

- **指轮式模块**

指轮式模块允许指轮式开关与 PLC 控制程序并行来处理反馈信息。指轮式信息通常为 BCD 码的形式，且允许人为的在外部改变设置点或预置点而不需要改变控制程序。

- **TTL 模块**

TTL 模块用来接收和传送与 PLC 处理器进行通信的 TTL 信号。TTL 电平信号具有处理器能够接收的形式，且需要缓冲区。

- **编码计数器模块**

编码计数器模块允许对增量式编码器和绝对值编码器进行连续监控。编码对轴的位置进行追踪。格雷码为常用的绝对值编码，通过对其解码可以实现定位。模块允许用户实时的从编码器中读入和存储信息，稍后再由处理器读取。

- **BASIC 模块或 ASCII 模块**

ASCII 模块允许传送和接收 ASCII 文件。这些文件通常为程序或操作数据。模块通常用 BASIC 命令进行编程。用户用 BASIC 语言进行编程。BASIC 模块用于向打印机或终端输出文本来更新操作。

- **步进电机模块**

步进电机模块向步进电机译码器提供一系列脉冲，以此来控制步进电机。模块的指令由 PLC 的控制程序决定。

- **BCD 输出模块**

BCD 输出模块能够使 PLC 对如七段显示器这样的，需要 BCD 码信号的设备进行操作。

由于一些特殊模块内具有与 PLC 并行处理能力的微处理器，因此把它们称为智能 I/O 模块。它们包括：

- **PID 模块**

比例－积分－微分（PID）模块用于包含 PID 算法的流程控制中。基于数学运算的算法是一个复杂的程序。PID 模块允许在 CPU 之外对流程进行控制。这种方式可以使 CPU 免于进行大量复杂的计算。PID 模块中的微处理器先对数据进行处理，然后与 CPU 提供的参考点进行比较，最后产生合适的输出信号。

- **伺服模块**

伺服模块用于闭环控制系统。闭环控制是通过引入从设备的反馈来完成的。模块的编程通过 PLC 完成；但是一旦编程完毕，模块就能独立的控制设备。

- **通信模块**

由于集成系统的不同，数据必须能够对所有系统共享。PLC 必须能够和计算机、计算机数控（CNC）机、机器人以及其他 PLC 进行通信。通信模块允许用户将 PLC 连接在高速局域网中，这可能与 PLC 提供的网络通信不同。

- **语言模块**

语言模块使用户能够用高级语言来编写程序。通过高级语言编译器，可以将高级语言

转换为PLC处理器能够识别的机器语言。BASIC是最常用的语言模块。还有其他可用的语言模块如C、Forth和PASCAL模块等。

• **语音模块**

语音模块通常用于将人发出的字、短语或句子进行数字化。数字化的声音被存储在模块存储器中。每个字、短语或句子都被赋予一个数字。然后在恰当的时间通过梯形图输出正确的信息。

为了满足特定的需要，新型的模块还在不断地开发出来。同时，有些模块已不再需要，如PID模块，这是由于更新的PLC中将PID功能包含在指令当中了。

2.5 I/O模块的技术指标

生产厂商的技术指标提供了如何正确安全使用接口设备的大量信息。技术指标不仅对可操作的模块，而且对现场设备做出了一定的限制。以下是一些常用的生产厂商提供的I/O技术指标，以及对它们的简介。

• **额定输入电压**

交流或直流额定电压值规定了可以接受的电压信号的大小及类型。

• **导通状态的输入电压范围**

导通状态的输入电压值规定了输入信号的电压是被认为处于绝对导通（ON）状态下的电压范围。

• **输入端额定电流**

输入端额定电流值规定了输入设备必须提供以能够驱动输入电路进行工作的最小电流值。

• **环境温度等级**

环境温度等级的值规定了能够使I/O模块处于最佳工作状态的最大温度值。

• **输入延迟**

输入延迟也被称为响应时间，它的值规定了在被确定为有输入信号到输入电路为ON的这段时间段。延迟是滤波电路为了保护触点跳动和电压瞬变所造成的。输入延迟的范围通常为9ms到25ms。

• **额定输出电压**

交流或直流额定电压值规定了能够被输出端控制的电压源电压的大小及类型。

• **输出电压范围**

输出电压范围规定了输出工作电压的最大值和最小值。例如，等级为交流120V的输出电路的工作范围可以为交流92V（最小值）到交流138V（最大值）之间。

• **输出端和模块的最大输出电流等级**

输出端和模块的最大输出电流等级规定了作为一个整体的单个输出端和模块能够使负载（额定电压下）安全工作的最大电流值。例如，规定值限制了每个输出端的输出电流不能超过1A。总模块电流的等级通常要小于单个模块的电流等级。由于8个设备当中的各个设备的电流通常不会同时为1A，因此总电流等级可以为6A。

• **最大输出冲击电流**

最大输出漏电流规定了能够超过输出电路最大连续电流值的最大冲击电流和时间（如：0.1s内电流为20A）。

• **输出端断开状态的漏电流**

输出端断开状态漏电流的值规定了在断开状态下流过输出端的最大漏电流的值。

• **电隔离**

电隔离定义了I/O电路和控制器逻辑电路之间隔离的最大电压值。通过隔离可以保护

模块的逻辑端，以免出现输入或输出端电压或电流过大的现象，但是模块的功率电路仍可能被损坏。

• **模块的 I/O 点数**

此数值表示能够连接到模块上的现场输入端和输出端的数量。有些模块提供了不止一个公共接线端，使得用户可将同一个插件用于不同的电压等级，以及可将电流进行有效的分配。

• **背板牵引电流**

此值显示了模块需要从背板获得的电流量。由一个机座上所有模块所牵引的背板电流总量用于选择合适电源等级的机座。

• **分辨率**

模拟 I/O 模块的分辨率规定了模拟值能够被数字值所表示的准确度。分辨率（通常用位来表示）越高，模拟值能被表示得越准确。

• **输入阻抗和电容**

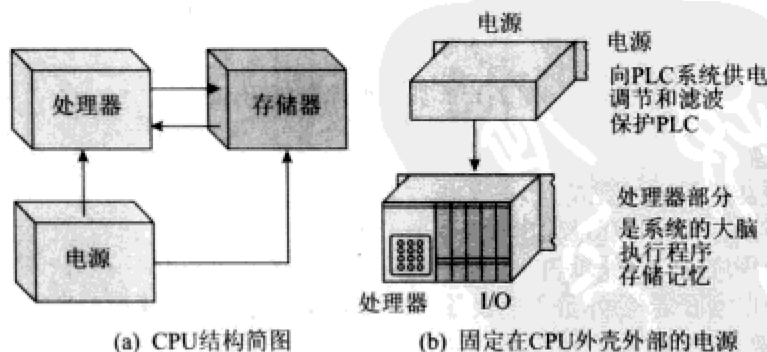
由于具有模拟 I/O 信号，输入阻抗值和电容值必须和与模块相连的外部设备匹配。常用等级为兆欧（MΩ）和皮法（pF）。

• **同相抑制比**

同相抑制比定义为模拟模块抑制噪声的能力，噪声通常对同一个信道或模块信道之间数据的完整性进行干扰。

2.6 CPU 部分

CPU 包括处理器-存储模块、通信电路和电源。图 2-14a 为 CPU 的示意框图。CPU 可以随着厂家的不同而具有不同的结构，但是大多数都按照图 2-14a 所示。电源可以安置在 CPU 外壳之内，也可以作为一个独立的单元固定于 CPU 之外，如图 2-14b 所示。电源单元同样具有一个备用电池组系统，这取决于存储器的类型为易失性存储器还是非易失性存储器。例如，在 Allen-Bradley SLC 500 系统中，电池组安装在处理器插件上。



(a) CPU结构简图　(b) 固定在CPU外壳外部的电源

图 2-14　CPU 的主要部件

PLC 的电源可以提供工作要求的所有等级的电压。电源将交流 120V 或 220V 转换为 CPU、存储器和 I/O 电子电路所需要的直流电压。PLC 工作在直流 -5V 到 +5V 之间。因此，电源必须能够将交流输入电压整流且降到可用的直流电压等级。

术语 CPU 通常和处理器交换使用。然而，按照严格定义，CPU 定义应包括构成智能系统的所有必要元件，而且在组成 CPU 的部分和在这些部分之中相互作用的常量之间具有明确的关系。处理器通过和系统存储器的连续作用来翻译和执行用于控制机器或流程的

用户程序。系统电源提供所有必要的电压等级，来确保所有处理器和存储器能够正常运行。

CPU 含有和个人计算机内部同样类型的微处理器。不同点就是微处理器所执行的程序是用来进行工业控制的，而个人计算机中的程序是提供通用计算的。CPU 能够执行系统操作、信息存储、输入监控、用户逻辑（梯形图）评价和正确地输出信号。PLC 系统中的 CPU 可以包含不止一个微处理器。使用多微处理器的优点是可将通信与控制任务分离，从而改善整体操作速度。例如，有些 PLC 厂家使用一个控制微处理器和一个逻辑微处理器。控制微处理器进行更复杂的计算和数据操作；逻辑微处理器执行定时、逻辑和计数工作，同时监控应用系统。

36

与处理器单元相结合的，还有许多可以将系统的诊断信息提供给操作员（如图 2-15 所示）的 LED 状态显示器。同样，键开关可以允许操作员任选以下三种工作模式之一（否则这些模式的选择必须从编程设备输入）：运行模式、编程模式和远程模式。

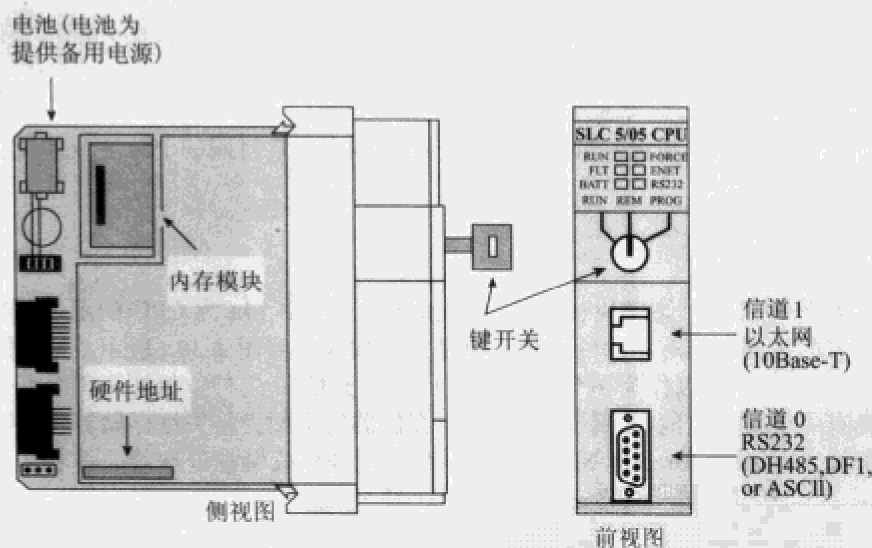


图 2-15　典型处理器单元

- **运行位置**

❑ 将处理器置于运行模式。
❑ 执行梯形图程序且驱动输出设备。
❑ 禁止对程序进行在线编辑。
❑ 禁止使用编程器/操作员界面设备来改变处理器的工作模式。

- **编程位置**

❑ 将处理器置于编程模式。
❑ 禁止处理器对程序进行扫描或执行，而且控制输出端失电。
❑ 允许输入和编辑程序。
❑ 禁止在使用编程器/操作员接口设备时改变处理器的工作模式。

- **远程位置**

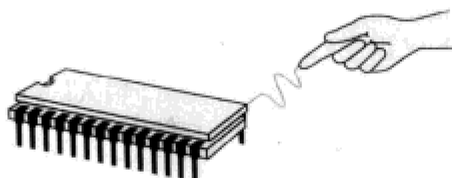
❑ 将处理器置于远程模式：远程运行模式、远程编程模式或远程测试模式。
❑ 允许利用编程器/操作员接口设备来改变处理器的工作模式。
❑ 允许在线编辑程序。

处理器模块也包括和编程装置通信的电路。在模块的某个部位有一个连接器，使 PLC 可以和外部编程设备相连接。

PLC 的决策能力已经超过了简单的逻辑处理能力。处理器能够执行其他的功能，如定时、计数、锁存、比较以及超越加、减、乘、除四种基本运算的复杂数学运算。

由于计算机技术的不断发展以及应用需求的不断加大，PLC 处理器也在不断发生变化。目前随着新的工作模式被引进来，处理器的执行速度比以前更快，且增加了新的指令。由于 PLC 的核心是微处理器，所以它能够执行计算机所能执行的任务。另外，PLC 除了原有的控制功能外，还能实现网上监控以及数据采集（SCADA）。

处理器或其他模块中的一些电子元件对静电压非常灵敏。静电电压可以低于 30V 且电流小于 0.001A，远小于人所能感觉、听到或看到的程度（如图 2-16 所示）。当静电电压达到 3500V 时，人才可以感觉到静电放电作用；达到 4500V 时，才可以听到放电的声音；达到 5000V 时，才可以看到放电产生的火花。工作台附近的人在正常运动的情况下能产生高达 6000V 的静电电压。在干燥的环境下，积累在走过尼龙地毯的人身上的静电压能达到 35 000V。当操作静态敏感设备或模块时，应当按照如下的静态控制步骤来执行：



静电放电能够产生损坏敏感设备或降低性能的电流

图 2-16　静电损坏

- 在操作静态敏感元件之前，人员需要接触导电表面使得自己接地。
- 戴上腕箍，可以提供一个将工作中产生的静电荷放掉的通道。
- 不要接触 PLC 系统背板上的接线器或接线器引线（通常对电路插件进行操作）。
- 当对模块内部的元件进行配置或替换时，不要接触模块上的其他电路元件。
- 通过一个接地的导体表面屏蔽工作台以及地面形成一个静态安全工作区。

2.7　存储器设计

存储器是 CPU 中用来存储和处理程序文件（控制表）和数据文件的一块物理空间。数据通常以地址形式存储在存储区中。存储在存储器中的信息和如何处理输入和输出数据相关。

程序的复杂性决定了所需的存储量。存储单元以位（二进制数）为单元对信息进行存储。规定存储容量以 1000 或 K 为增量，1KB 为 1024 字节（1 字节为 8 bit）。

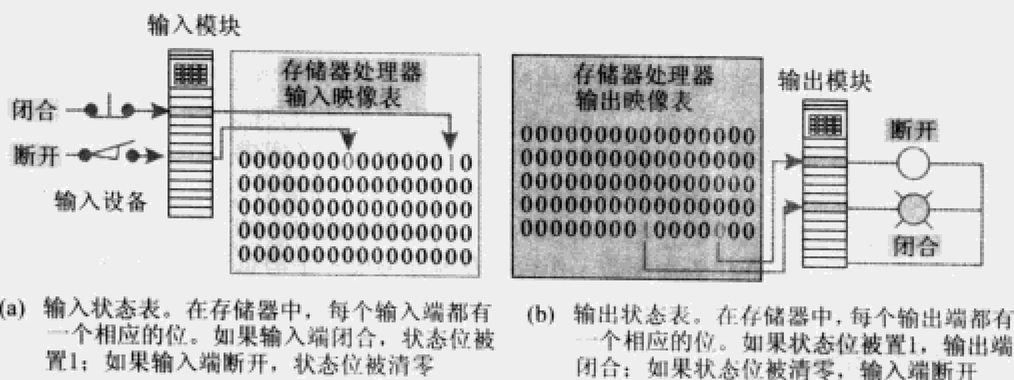
程序以 1 和 0 的形式存储在存储器中，这些 1 和 0 以 16 位字的形式进行汇编。存储器的大小通常以能够被储存在系统中的千个字来表示；因此 2KB 表示能存储 2000 个字的存储器，64KB 表示能存储 64 000 个字的存储器。存储器的大小的变化范围可以从 1KB 的小系统到 2 000K（2M）B 的大系统。下表显示了一部分 Allen-Bradley 生产的 PLC 的性能规格的概述。

PLC 类型	存储器	I/O 点	通信选项
MicroLogic 1000	1KB	小于 20 个输入点 小于 14 个输出点	串行，DH-485，设备联网
SLC 500	小于 64KB	小于 4096 个输入点和输出点	串行，DH-485，DH +，设备联网，控制网络，以太网
ControlLogix	160KB 到 2MB	小于 128 000 个输入点和输出点	串行，DH-485，DH +，设备联网，控制网络，以太网

存储单元表示 CPU 存储器中能够存储一个二进制字的地址。一个触点也可以使用存储器中的一个单元。能够存储在 PLC 的 RAM 存储器中的总位数称为存储器容量。存储器容量大多数以字节表示。一个字可以是 1 个、2 个或 4 个字节。

存储器利用率表示需要用来存储每种类型指令的存储单元数。关于存储器单元的一条经验法则是每个线圈或触点用一个单元。1KB 的存储器允许程序包含 1000 个线圈和触点存储在 RAM 中。如果一个定时器占据两个字节，在 1KB 的存储器中可以包含 512 个定时器。

PLC 的存储器可以分为具有特定功能的几部分。存储部分用于储存输入端和输出端的状态，通常称为输入状态存储区或输入状态表，以及输出状态存储区或输出状态表（如图 2-17 所示）。这些术语简单的表达了存储输入设备或输出设备状态的一个单元。每个位可以是 0 也可是 1，这取决于输入端触点是打开还是闭合。闭合触点的状态将以二进制数 1 储存在相应输入表的存储单元中，而断开触点的状态将以 0 进行存储。当一盏灯发光时，其状态将以 1 存储在它相应输出表的单元中，相反当灯熄灭时，其状态以 0 进行存储。输入和输出映像表通过 CPU 不断进行修正。每次存储单元被检查时，如果触点或线圈的状态发生改变，映像表将被更新。



(a) 输入状态表。在存储器中，每个输入端都有一个相应的位。如果输入端闭合，状态位被置1；如果输入端断开，状态位被清零

(b) 输出状态表。在存储器中，每个输出端都有一个相应的位。如果状态位被置1，输出端闭合；如果状态位被清零，输入端断开

图 2-17 输入和输出状态表和存储区

39

状态表存储区用于存储系统信息，如扫描时间、故障状态、故障编码和看门狗定时器；还有用于程序控制的精确定时位。定时器存储区通常为三个字的长度。其中一个字表示定时器的状态信息；第二个字表示预置值或调整点；最后一个字则是累加值。计数器存储区同样具有三个字的长度，它的结构和定时器存储区一样。位存储区、控制存储区和整数存储区都是用来更灵活地编程和使用更复杂的指令。

虽然存储器有许多类型，但是通常划分为两类：断电保持和非断电保持。如果所有工作电源消失或被去掉，非断电保持型存储器将失去它所存储的信息；非断电保持型存储器容易发生变化，且十分适合于具有备用电池的应用。

当突然失电或瞬时失电时，非断电保持型存储器能够保持所储存的信息。虽然非断电保持存储器一般为不可改变的，但存储在有些特殊类型存储器中的信息可以被改变。

为了确定 PLC 存储器没有被损坏，需要 PLC 执行存储器检查。由于安全因素，需要对存储器进行检查。这将保证在存储器损坏的情况下 PLC 不执行程序。程序通常以最快的速度在 RAM 中运行，且在上电时由 EEPROM 或 EPROM 传送到 RAM。

2.8 存储器类型

PLC 具有允许用户开发和修改控制程序的可编程存储器。这种存储器为非断电保持

型，以至于当PLC失电时，程序不会丢失。PLC可以利用不同类型的存储器设备。

数据存入处理器存储单元被称为写入。数据读出的方式可以从存储器中读取程序。以下是对常用存储器类型总的描述。关于特殊存储器类型的详细信息，可以从控制器的软件包部分所提供的说明书中获得。

2.8.1 只读存储器（ROM）

存储于只读存储器（ROM）中的信息只能够被读取，并且在通常条件下不能够进行改动。信息是由厂家植入存储器中，为PLC的内部使用和操作。只读存储器为非易变型存储器；当电源消失时，它们能保持信息不丢失且不需要备用电源。

ROM用于PLC的操作系统。PLC生产商将操作系统植入ROM，且控制被用户用来对PLC编程的系统软件。编程器产生的梯形图语言是一种高级语言。操作系统软件必须将梯形图程序转换为微处理器能够识别的指令。

2.8.2 随机存取存储器（RAM或R/W）

随机存取存储器通常也定义为读写（R/W）存储器，信息能够写入或从存储器中读出。RAM是一种固态的存储器，它被包含在集成电路上，且通常用于用户存储器。用户程序、定时器/计数器的值、输入/输出状态等信息都存储在RAM中。当程序被执行时，处于程序控制下的微处理器允许向RAM输入信息，从而使存储的信息发生改变。

40

如果电源消失，RAM中的数据也会丢失。易变型RAM通常用电池来保护（如图2-18所示）。如果电源暂时消失，PLC不会丢失其程序，可通过一个后备电池在电源断电时，备用电源继续为PLC供电。当PLC被关掉时，备用电源开始起作用。大多数PLC在用户存储器中使用了CMOS-RAM技术。CMOS-RAM芯片具有很小的驱动电流且能够保持程序不丢失，其中使用了一个供电时间可长达两年到五年的锂电池。当电源消失且电池没有装上时，有些处理器利用一个电容来等效为一个可以持续至少30 min的后备电池。在这30 min内，RAM中的数据不会丢失。

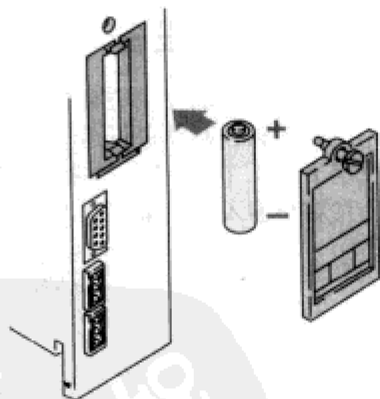


图2-18 用于保护RAM的电池

很多控制器在用户程序存储器中都使用了由电池供电的CMOS-RAM芯片。RAM中创建和修改程序非常方便。由于CMOS-RAM芯片在没有被存取时的驱动电流非常低（15μA），且只需要提供2V的直流电压，存储在里面的信息就能保存，因此它的使用范围非常广。

2.8.3 可编程只读存储器（PROM）

可编程只读存储器（PROM）是特殊类型的ROM。可编程只读存储器允许将内部和/或外部信息写入芯片中。在出厂后，可编程只读存储器只能被写入信息一次。当程序输入完毕，脉冲电流熔断芯片中的熔断丝，可以防止芯片被重新编程。只有极少数的控制器在程序存储器中使用PROM，这是因为任何程序的改动都将导致更换新的芯片。

2.8.4 可擦除可编程只读存储器（EPROM）

可擦除可编程只读存储器（EPROM）是一种特殊的PROM，它能够利用一个紫外光源将程序完全擦除后重新进行编程。通常也称为紫外光PROM（UVPROM），此芯片集成电路的硅材料上具有一个石英窗。此窗口通常由遮光涂料所掩盖。当移去遮光涂料时，电

路将暴露在紫外光的照射下，20 min后，存储内容将完全被擦除。一旦存储内容被擦除后，可以使用编程设备对EPROM芯片进行重新编程。

EPROM或UVPROM用于备份、存储或传递PLC程序。PLC处理器只能从这类存储器设备中读入数据。外部PROM编程器用于对设备进行编程（写入数据）。UVPROM是断电保持型存储器设备且不需要备用电池。

2.8.5　电可擦除可编程只读存储器（EEPROM）

电可擦除可编程只读存储器（EEPROM）是断电保持型存储器，它能像RAM一样灵活地进行编程。EEPROM能够用电写入新的数据而不需用紫外线擦除存储器。由于EEPROM是断电保持型存储器，因此它不需要预备电池。它能永久地存储程序，并且可以利用标准编程设备方便地更改程序。通常EEPROM模块用于存储、备份或传递PLC程序。

41

近来，PLC的处理器开始使用快速存储器。快速EEPROM和EEPROM类似，都只能用于备份存储器。主要的差别就体现在"快速"：在文件存储和读取方面格外迅速。另外，在重新编程时，快速存储器不需要从处理器上移开；可以通过使用处理器中的电路来完成重新编程。快速存储器有时也是位于CPU模块内，它能自动为RAM备份。如果一个带有快速存储器的PLC在运行时突然断电，当电源恢复之后，PLC将继续运行而不会丢失任何工作数据。

2.9　编程设备

编程设备的使用方便是PLC的一个重要特征。通过编程设备提供的这些基本方式，用户可以和控制器进行通信（如图2-19所示）。编程设备用于输入所需的指令。这些指令决定了PLC在一定输入的情况下会做出什么决策。操作员界面允许对显示的信息进行处理且对新输入的参量控制。

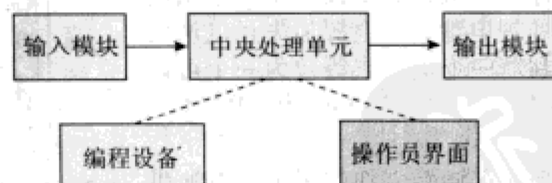


图2-19　用户和PLC电路之间的通信

PLC厂商使用各种类型的编程单元来输入程序和数据。最简便的专有编程单元为手持编程器（如图2-20所示）。手持编程设备具有一根连接电缆，可以很方便地插入PLC的编程端口。手持编程器结构紧凑，价格低廉，且使用方便。手持编程器的组成部分包括多种颜色和多种功能的按键和一个液晶显示器（LCD）或发光二极管显示窗（LED）。常用按键用于输入和编辑指令，定位键用于程序移动。手持编程器的显示能力有限。有些手持编程器只能显示最近输入的一条指令，而有些能显示两个或四个梯级。有些微控制器也直接使用插入式板而不使用手持设备。

手持编程设备有两种类型：非智能终端型和智能终端型。非智能终端型编程器没有内嵌的智能设备或存储器。为了编写、修改或监控程序，编程器必须连接到PLC上。多数新型手持编程终端都是智能型。智能型手持编程终端具有自己的微处理器，它能够独立于PLC进行工作。灵活型编程终端有时也称为独立终端，它不用和PLC进行连接而在离线

状态下编写程序。当连接到 PLC 时，通信线路能够用于从控制器 CPU 的智能终端下载程序。同样地，如果需要查看 PLC 中正在运行的程序的复制本，可以将复制本上传到灵活手持编程器的存储器中。

42

PLC 生产商曾经卖过专用编程器，它除了只用于编程和监控 PLC 外，其余看起来和个人计算机很相似。这些类型的编程器现在已经很少看到。目前，最流行的编程方式是利用个人计算机在 DOS 或 Windows 模式下运行生产商提供的软件来对特定的 PLC 编程（如图 2-21 所示）。使用个人计算机编程具有以下优点：

- 大量的逻辑信息可以显示在监视器上，能够容易地译读程序。按住向上或向下方向键可以方便地由一个梯级滚动到另一个梯级。
- 彩色监视器能够以不同的颜色来显示电路元件的状态。
- 可以将一个或多个程序存储在电脑的硬盘驱动器中。
- 计算机可用于记录 PLC 程序。记录文档的形式可能采用为在维护和检修中可能用到的元件和注释作标签的。
- PC 软件为程序的编写和修正提供了剪切和粘贴功能。
- PC 能方便地监控数据表。
- 在软盘、CD-ROM 或硬盘驱动器上可以方便地进行程序的复制。
- 有些软件包中额外的图解能力能开发受控设备的流程图。
- 便携式或笔记本 PC 体积非常小，且非常轻便。

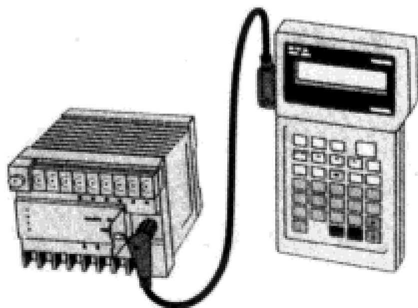


图 2-20 手持编程设备

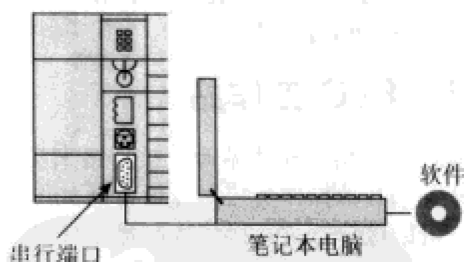


图 2-21 使用个人计算机编程

2.10 数据存取

打印机用于以梯形图形式提供处理器存储器的硬件复制打印输出。过长的梯形图不能完全在屏幕上显示出来。通常，显示屏一次性最多显示五个梯级。打印输出能够显示任何长度的程序，来分析完整的程序。

有些老的 PLC 系统利用盒式磁带记录器来记录和存储用户程序。这些类型的系统已经被电脑磁盘驱动器所取代。利用软盘来记录和存储程序的优点是：存取速度快，程序访问迅速和数据存储量大（如图 2-22 所示）。

43

在存储器中，PLC 每次只允许存在一个程序。要改变 PLC 中的程序，必须直接从键盘重新输入新的程序或从电脑的硬盘上下载程序。有些 PLC 使用内部 EEPROMS 或 EPROM 存储器模块（如图 2-23 所示），这些模块能够对 PLC 中的程序进行备份。如果 PLC 丢失了它内部的程序，在 EEPROM 或 EPROM 中的程序将被迅速下载到 PLC 的存储器中。

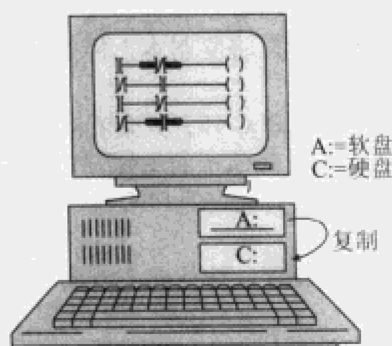


图 2-22 程序复制

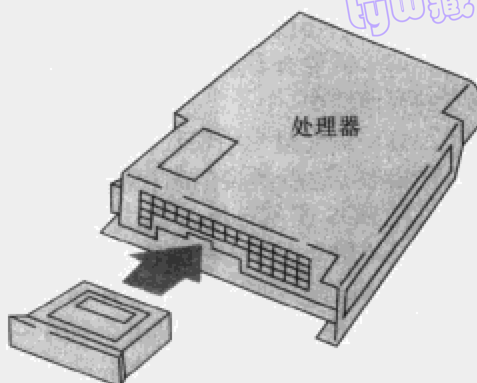


图 2-23 安装在处理器上的 EEPROM 或 UVPROM 存储器模块用于对 PLC 中的程序进行备份

由于有了存储器模块，可以实现：

- 存储处理器 RAM 中的内容。
- 将 EEPROM 和 EPROM 存储器中的内容载入处理器 RAM 中。
- 当程序需要保密时，可以使用 EPROM 存储器模块，因为当模块被安装在控制器内部时，在 EPROM 中的程序不能更改。
- 配置 PLC 使之能在上电的情况下自动下载程序，或如果存储器出现故障的情况下能自动下载程序。

2.11 PLC 工作站

44

PLC 工作站或操作界面可被连接来实现和 PLC 之间的通信，且能取代按钮、指示灯、指轮开关和其他操作员控制盘设备（如图 2-24 所示）。荧光触摸屏小键盘提供了一个操作界面，可以像传统的硬接线控制盘一样进行操作。

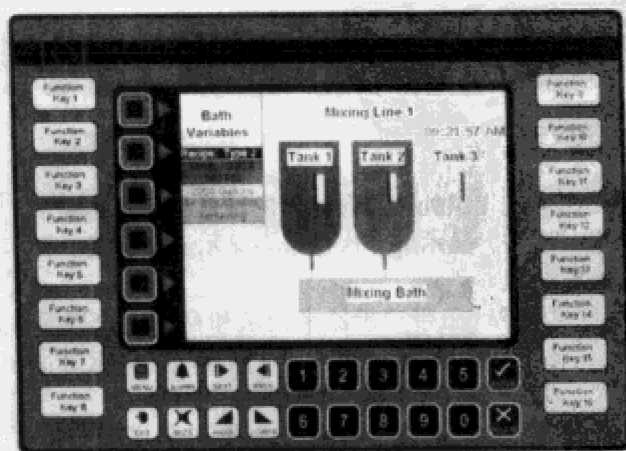


图 2-24 常用的 PLC 操作界面（得到位于美国宾夕法尼亚州约克郡的 Red Lion Controls 公司许可）

通过在个人电脑上设置软件，可以对显示屏进行配置以实现如下功能：

- 可以用逼真的插图来替换硬接线的按钮和指示灯。机器操作员只需要触摸显示盘就可触发按钮。
- 以图解的方式能够更容易地显示动作。
- 允许操作员通过触摸显示屏上的数值小键盘图片来改变定时器和计数器的预置值。
- 显示报警，并记录报警出现的时间和位置。
- 显示变量随着时间变化的过程。

45

思考题

1. PLC 输入模块的功能是什么？
2. PLC 输出模块的功能是什么？
3. 描述术语逻辑机架。
4. a. 什么是远程机架？ b. 为什么需要远程机架？
5. 处理器怎样识别具体输入或输出设备的位置？
6. 描述 I/O 地址的三个基本因素。
7. I/O 模块的接线端以什么方式进行连接？
8. 比较标准 I/O 模块与高密度 I/O 模块。
9. 什么类型的现场输入设备需使用离散输入模块？
10. 什么类型的现场输出设备需使用离散输出模块？
11. 列出 I/O 模块电路中光电隔离电路的三种功能。
12. 写出 I/O 模块的两个基本部分的名称。
13. 列出输入模块执行的四种任务。
14. 在 120V 交流输出接口模块中通常使用什么电子元件作为开关元件？
15. a. 常用 120V 交流输出接口设备的最大电流等级是多少？
    b. 解释当输出信号需要大电流时，需要进行怎样的处理。
16. 直流输出模块常用什么电子元件作为开关元件？
17. 继电器型输出模块可用于控制什么类型的负载？
18. 比较电源现场设备和吸收现场设备的连线方式。

46

19. a. 比较离散 I/O 模块和模拟 I/O 模块所适用的输入设备或输出设备的类型。
    b. 解释模拟输入电路中 A/D 转换电路的功能。
    c. 列出三种常用模拟输入传感设备的类型。
    d. 为什么在低压模拟传感设备的接线中通常使用屏蔽电缆？
20. 陈述如下各种特殊 I/O 模块的一种运用：
    a. 高速计数模块 b. 指轮式模块 c. TTL 模块 d. 编码器计数器模块 e. BASIC 或 ASCII 模块 f. 步进电机模块 g. BCD 输出模块
21. 列出四种带有微处理器的智能 I/O 模块。
22. 对如下各 I/O 参数做一个简短的描述：
    a. 额定输入电压 b. 导通状态输入电压 c. 输入端的额定电流 d. 额定输出电压 e. 输出电压范围 f. 最大输出电流等级 g. 断开状态输出端漏电流 h. 电气隔离 i. 输入输出点数 j. 背板牵引电流 k. 分辨率 l. 输入电阻和输入电容 m. 同相抑制比
23. 描述 CPU 三个主要部分的基本功能。
24. 列出三种可以通过处理器单元的按键开关来进行选择的典型工作方式。
25. 除了简单的逻辑处理外，陈述 PLC 处理器能够执行的其他三种功能。
26. 为了防止损坏对静电敏感的 PLC 元件，需要采取什么措施？

47

27. a. 在输入状态表和输出状态表中存储的是什么信息？
    b. 这些信息是怎样存入存储器中的？

28. 比较非断电保持型存储器和断电保持型存储器存储的特点。
29. 为什么 PLC 要进行存储检查?
30. 考虑如下因素，比较 ROM 和 RAM 的设计：
    a. 信息植入存储器中的方式 b. 改变存储器信息的方式 c. 按易失型和非易失型分类
31. a. 怎样将初始信息和/或额外信息写入 PROM 芯片?
    b. PROM 存储器芯片的主要限制是什么?
32. 怎样擦去如下芯片中的程序：
    a. EPROM b. EEPROM
33. 讨论使用快速存储器的处理器的优点。
34. 列出 PLC 编程设备的三种功能。
35. 列出三种适用的编程设备。
36. 将非灵活型和灵活型手动编程终端进行比较。
37. 怎样将个人电脑作为一个 PLC 的编程器使用?
38. 程序的电脑文档中包含什么信息?
39. 使用打印机将程序打印出来有什么好处?
40. 与磁带存储器相比，列出三种使用软盘来记录和存储 PLC 程序的优点?
41. 解释安装在处理器上的 EEPROM 或 UVPROM 存储器模块的功能。
42. 列出 PLC 工作站能够完成的 7 种功能。 [48]

习题

1. 离散交流 120V 输出模块用于控制直流 230V 螺线管操作阀。画出框图显示怎样使用插入式继电器来完成控制。
2. 假设当温度在 750°F 到 1250°F 范围变化时，热电偶能产生 20mV 到 50mV 之间变化的线性电压。当温度为 1000°F 时，热电偶将产生多大的电压?
3. a. 给定模块的输入延迟时间规定为 12ms。以秒来表示该为多少时间?
   b. 给定模块的输出漏电流规定为 950μA。以安培来表示电流值该为多少?
   c. 给定 I/O 模块的最大环境温度规定为 60℃。以华氏温度来表示温度值该为多少?
4. 按照以下要求构造一个典型的五位制地址（根据图 2-4）：
   a. 按钮与位于机架 1 上模块组 2 的接线端 5 相连接。
   b. 灯与位于机架 2 上模块组 0 的接线端 3 相连接。
5. 假设交流输出模块的双向三极管开关由于短路而失效。这将会给连接在输出端的设备带来什么影响?
6. 个人计算机用于编程几个不同的 PLC 模型，有什么要求? [49]

第3章　数制与码制

学习目标：

- ❑ 定义十进制、二进制、八进制和十六进制数制系统，且能将一种数制或编码系统转换为另一种数制或编码系统。
- ❑ 解释说明 BCD 码、格雷码和 ASCII 码系统。
- ❑ 通过在二进制存储单元中的应用，定义术语比特、字节、字、最低有效位（LSB）和最高有效位（MSB）。
- ❑ 二进制数的加、减、乘和除。

使用 PLC 需要对除了十进制数制系统之外的其他数制系统也非常熟悉。有些 PLC 模块和单个 PLC 功能需要使用其他的数制系统。本章对这些数制系统进行了介绍，包括：二进制、八进制、十六进制、BCD 码、格雷码和 ASCII 码。在此基础上，讲述了各种进制之间的相互转换。

3.1　十进制系统

当使用 PLC 或各种数字计算机时，不同数制系统和数码的知识非常有用。这是由于这些设备是通过数字来进行表达、存储或操作。通常，PLC 是对二进制数进行操作，用二进制数来表示变量或变化的码值。

最常用的十进制系统是以 10 为基值。数制系统的基数或基值决定了用于系统中不同符号或数字的总数。例如，在十进制系统中，具有 10 个独特的数字——数字 0 到 9；符号的总数和基值一样，且最大值为基值减 1。

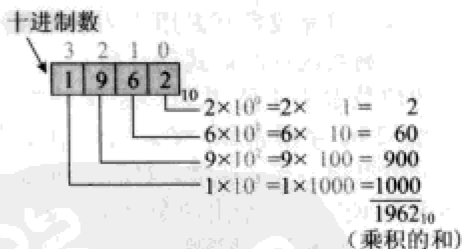


图 3-1　十进制系统中的权值

十进制数的值，取决于它的位数及每位的值。分配（权）值被赋值到各个位置，数字是按从右到左的方式置于位置中。在十进制系统中，从最右边起的第一位为 0；第二位为 1；第三位为 2；一直持续到最后一位。每个位的权值能够通过基值（在此例中为 10）及其幂来表示。例如十进制系统中，位置权值为 1、10、100、1000 等等。图 3-1 说明了怎样通过将位置上的权值进行乘法操作和加法操作来得到十进制数值。

3.2　二进制系统

二进制系统以数字 2 为基数。此系统中只允许有数字 0 和 1。在数字电路中可以很容易地判别出和二进制数 1 和 0 相关联的两种不同的电压（如 +5V 和 0）（如图 3-2 所示）。因此二进制系统能够非常方便地用于 PLC 和计算机系统。大多数 PLC 定时器和计数器都工作在二进制模式下。

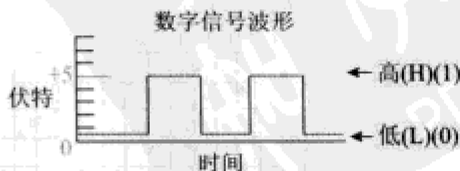


图 3-2　数字信号波

由于二进制系统只具有两种数字，因此二进制数的各位置只有两种变化，然后左边接近的位增 1。表 3-1 对四种常用数制系统进行了比较说明：十进制（基数 10）、八进制（基数 8）、十六进制（基数 16）和二进制（基数 2）。

注意，所有数制系统都是从 0 开始。

表 3-1 数制系统的比较

十进制	八进制	十六进制	二进制	十进制	八进制	十六进制	二进制
0	0	0	0	11	13	B	1011
1	1	1	1	12	14	C	1100
2	2	2	10	13	15	D	1101
3	3	3	11	14	16	E	1110
4	4	4	100	15	17	F	1111
5	5	5	101	16	20	10	10000
6	6	6	110	17	21	11	10001
7	7	7	111	18	22	12	10010
8	10	8	1000	19	23	13	10011
9	11	9	1001	20	24	14	10100
10	12	A	1010				

51~52

与二进制数等价的十进制数的值的计算方法和十进制数的计算方法相似。在这种情况下，权值为 1、2、4、6、8、16、32、64 等。二进制中，幂位置的权值不再为 10，而是 2。图 3-3 说明了怎样将二进制数 10101101 转换为与其等价的十进制数 173。

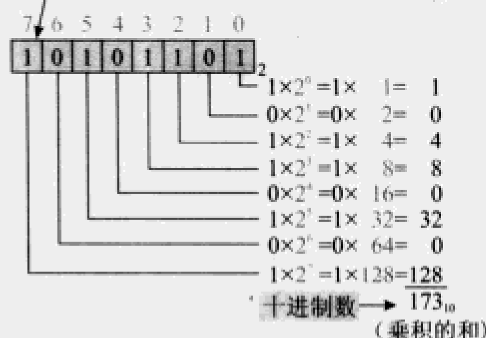


图 3-3 二进制数转换为十进制数

二进制数的每个数字都称为一个位。在 PLC 中，处理存储部分由成百上千的单元组成。这些单元或寄存器用字表示。每个字能够以二进制数或位的形式储存数据。一个字所包含的位数取决于所用 PLC 系统的类型。16 bit 字和 32 bit 字最常用。在一个字中，位也能通

53

过分组得到字节。通常，具有 8 bit 的组称为一个字节，具有 2 个或多个字节的组称为字。图 3-4a 说明了一个由两个字节组成的 16 bit 字。最低有效位（LSB）为代表最小值的数字，最高有效位（MSB）为代表最大值的数字。字中的位只存在两种状态：逻辑 1（或开）状态，或逻辑 0（或闭）状态。

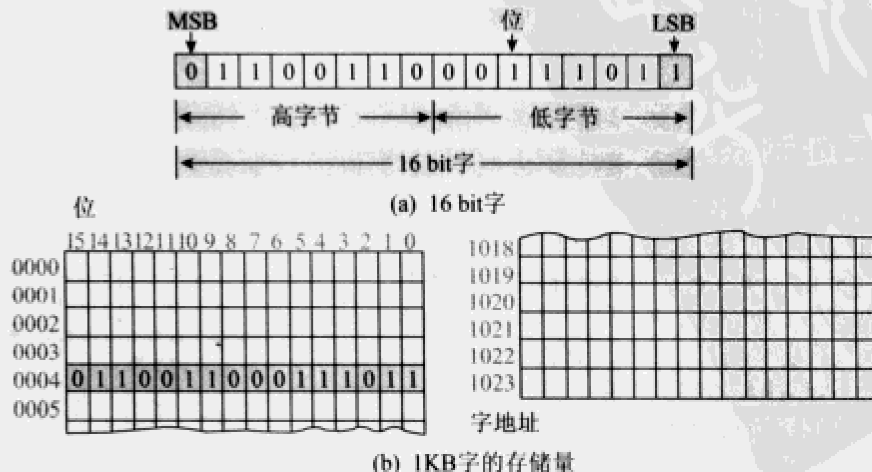


图 3-4 PLC 处理器的字和存储量大小

可以将数据分成两个 8 bit 的数据组来存储在 16 bit 字中。在低位字节中包含数据的 8 bit，同时在高位字节中包含数据的另外 8 bit。在较新一代的 PLC 中使用了 32 bit 的字，如 Allen-Bradley ControlLogix PLC。32 位的字也称为双整型数。

PLC 存储器的大小和能够存储的用户程序的数量相关。如果存储器的大小为 1KB 字（如图 3-4b 所示），如果它是 16 bit 字信息的，则能存储 1024 字或 16 384（1024 × 16）bit，如果它是 32 bit 字信息的，则能存储 32 768（1024 × 32）bit。目前的 PLC 包含的存储器大多数为 RAM，其大小可以从 1KB 字到 256KB 字。

为了将十进制数转换为对应的二进制数，必须连续地用 2 来除所需转换的十进制数。图 3-5 说明了十进制数 47 转换为二进制数的过程。首先将 47 除以 2。如果存在余数，将余数作为二进制数的最低有效位。如果没有余数，将 0 作为最低有效位。得出一次除法的结果后，处理器不断重复除法过程，直到累次除法的结果为 0。

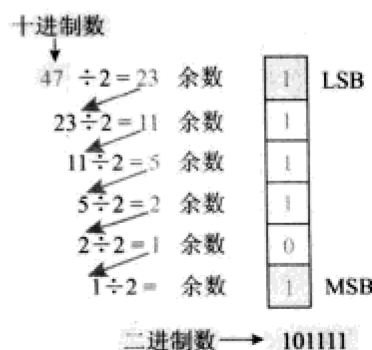


图 3-5 十进制数转换为二进制数

即使二进制系统中只具有两种数字，但它能用于表示十进制系统中的任何数。所有 PLC 的内部都工作在二进制系统中。处理器作为一个数字设备，只能识别 0 和 1，或二进制数（通常也叫机器语言）。例如，将梯形图程序送入译码器或汇编器中，把程序转换为处理器能够识别的机器语言。

计算机存储则是一串二进制数 1 和 0。图 3-6 显示了用于 Allen-Bradley SLC-500 模块机架的输出状态文件，它由组成 16 bit 字的单个位构成。一个 16 bit 输出文件的字用于表示机架上的一个槽。每个位代表一个输出点的开或闭的状态。这些点在顶排由右向左，从 0 到 15 进行编号。右边的列列出了输出模块的地址。虽然表中显示了顺序编址的输出状态文件的字，但实际上仅当处理器发现特殊的插槽中存在一个输出模块时，表中才产生一个字。如果插槽是空的，将不会在表中产生任何字。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Address
1	1	0	0	0	0	1	0	1	1	1	1	0	0	0	1	O:1
0	0	1	1	0	1	0	0	0	0	0	0	1	1	1	1	O:2
1	0	1	0	1	1	0	0	1	1	1	0	0	0	0	1	O:3
0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	O:4
1	1	1	0	1	0	0	1	1	1	0	0	1	1	0	1	O:5

图 3-6 SLC 500 输出状态文件

3.3 负数

如果一个十进制数为正，它具有一个正号；如果为负，它具有一个负号。在用于 PLC 的二进制数系统中，不可能用正号或负号来表示一个数的极性。一种表示二进制数是正还是负的方法是在数的最高有效位使用附加位或符号位。在符号位中，0 表示这个数为正，1 表示这个数为负（如表 3-2 所示）。

表 3-2 带符号的二进制数

符号	量值	十进制值	符号	量值	十进制值	符号	量值	十进制值
0	111	+7	0	010	+2	1	011	-3
0	110	+6	0	001	+1	1	100	-4

（续）

符号	量值	十进制值	符号	量值	十进制值	符号	量值	十进制值
0	101	+5	0	000	0	1	101	-5
0	100	+4	1	001	-1	1	110	-6
0	011	+3	1	010	-2	1	111	-7

在数字系统中，另一种表示一个负数的方法是使用二进制数的补码。为了补充一个二进制数，要将所有的1变为0，且将所有的0变为1。这是二进制数的第一种补码形式。例如，1001的第一种补码形式为0110。

表示一个二进制数的最常用的方法是使用第二种补码。第一种补码加1所得的结果就为第二种补码。此数制系统如表3-3所示。符号位为0表示一个正数，而符号位为1表示一个负数。

表3-3　正数和负数的第一种补码和第二种补码表现形式

带符号的十进制数	第一种补码	第二种补码	带符号的十进制数	第一种补码	第二种补码
+7	0111	0111	-1	1110	1111
+6	0110	0110	-2	1101	1110
+5	0101	0101	-3	1100	1101
+4	0100	0100	-4	1011	1100
+3	0011	0011	-5	1010	1011
+2	0010	0010	-6	1001	1010
+1	0001	0001	-7	1000	1001
0	0000	0000			

使用第二种补码可以使PLC执行数学运算变得非常容易。通过形成第二种补码，可以生成正确的符号位。如果最高有效位为1，PLC就会知道从存储器中取出的数为一个负数。当从键盘输入一个负数时，PLC以第二种补码的形式储存此数。初始数字先经过第一种补码形式变换，再经过第二种补码形式变换，最后得到相应的十进制数。

3.4　八进制系统

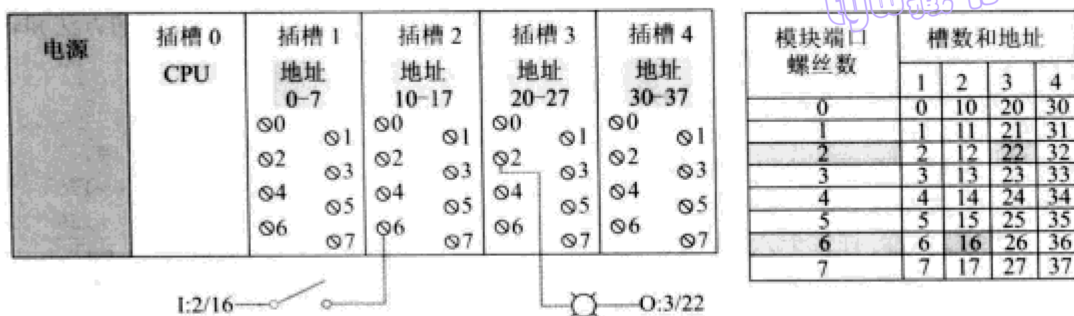
55

在二进制系统中表示一个数和在十进制系统中表示一个数相比，前者需要更多的数字。太多的二进制数使得阅读和书写变得非常不方便。为了解决这个问题，可以使用另一种相关的数制系统。

八进制系统是基数为8的系统，它通常用于微处理器、计算机和可编程控制系统，这是由于PLC用户或程序员可以利用组成一个信息字节中的8个数据位进行编址。图3-7说明了用八进制数制系统来对I/O模块编址。数字的范围从0到7；因此，不允许出现数字8和9。针对I/O数据，Allen-Bradley系列的PLC使用如下编址图表：

控制器	I/O编址
PLC-5处理器	基数为8（八进制）
SLC-500处理器	基数为10（十进制）
Logix控制器	基数为10（十进制）

八进制是一种处理较大的二进制数的方便方法。如表3-4所示，一个八进制数能够用三个二进制数字表示。和在所有其他数制系统中一样，八进制数中的每个数字根据其位置具有十进制数表示的权值。图3-8显示了怎样将八进制数462转换为相应的十进制数306。



模块端口螺丝数	1	2	3	4
0	0	10	20	30
1	1	11	21	31
2	2	12	22	32
3	3	13	23	33
4	4	14	24	34
5	5	15	25	35
6	6	16	26	36
7	7	17	27	37

图 3-7 使用八进制数制系统的 I/O 模块编址

表 3-4 二进制和与其相关的八进制码

二进制	000	001	010	011	100	101	110	111
八进制	0	1	2	3	4	5	6	7

56

八进制系统在 PLC 系统中很常用，这是因为八进制数能够非常容易地转换为二进制数。例如，如图 3-9 所示，通过汇编成三位数组，将八进制数 462 转换为相应的二进制数。注意计数法的简单性：八进制数 462 和它对应的二进制数相比，非常容易阅读和书写。

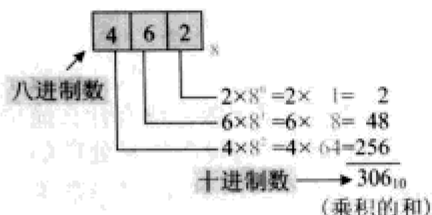


图 3-8 八进制数转换为十进制数

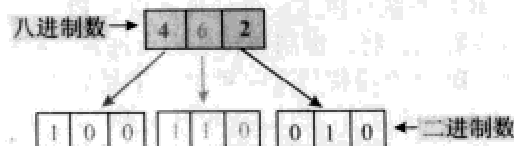


图 3-9 八进制数转换为二进制数

3.5 十六进制系统

由于一个数据的字由 16 个数据位或两个 8 bit 字节组成，因此十六进制（hex）数制系统可以用于 PLC。十六进制系统的基数为 16，用 A 到 F 来代表十进制数 10 到 15（如表 3-5 所示）。十六进制数制系统允许在一个如电脑屏幕或 PLC 编程设备显示器上小的空间里表现大量二进制位的状态。

表 3-5 十六进制系统

十六进制	二进制	十进制	十六进制	二进制	十进制
0	0000	0	8	1000	8
1	0001	1	9	1001	9
2	0010	2	A	1010	10
3	0011	3	B	1011	11
4	0100	4	C	1100	12
5	0101	5	D	1101	13
6	0110	6	E	1110	14
7	0111	7	F	1111	15

当将十六进制数转换为十进制数或将十进制数转换为十六进制数时，转换的方法和转换为二进制数及八进制数的方法一样。通过将十六进制数中各个位上的数字和对应的16的权重相乘，将十六进制数转换为相应的十进制数，这取决于数字的权。图3-10显示了怎样将十六进制数1B7转换为十进制数。

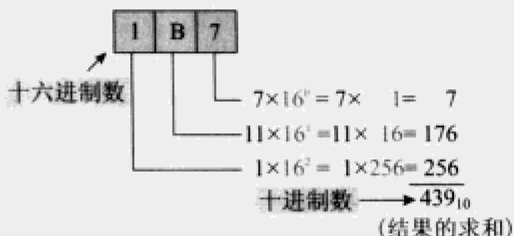


图3-10 十六进制数转换为十进制数

和八进制数一样，十六进制数转换为二进制数也非常容易。如图3-11所示，通过将十六进制数每位相应的4位二进制数字写入各个位置上就可完成转换。如图3-10和图3-11所示，十六进制数1B7的二进制数形式为000110110111，十进制数的形式为439。

57

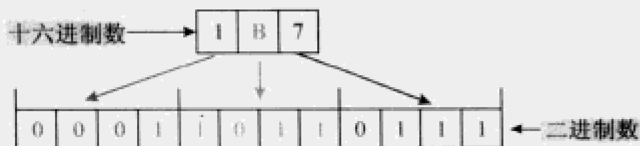


图3-11 十六进制数转换为二进制数

3.6 BCD系统

BCD系统提供了一种处理需要从PLC输入或输出大数字的便利方法。从各种数字系统可以看出，将二进制数转变为十进制数或将十进制数转变为二进制数，没有什么简单的方法。BCD系统提供了一种将人工易处理的码值（十进制）转换为设备能易处理的码值（二进制）的方法。PLC指轮开关和LED显示器就是PLC设备利用BCD数字系统的例子。表3-6显示了在十进制、二进制、BCD和十六进制中表示的数值。

表3-6 在十进制、二进制、BCD和十六进制中表示的数值

十进制	二进制	BCD	十六进制
0	0	0000	0
1	1	0001	1
2	10	0010	2
3	11	0011	3
4	100	0100	4
5	101	0101	5
6	110	0110	6
7	111	0111	7
8	1000	1000	8
9	1001	1001	9
10	1010	0001 0000	A
11	1011	0001 0001	B
12	1100	0001 0010	C
13	1101	0001 0011	D
14	1110	0001 0100	E
15	1111	0001 0101	F
16	1 0000	0001 0110	10

（续）

十进制	二进制	BCD	十六进制
17	1 0001	0001 0111	11
18	1 0010	0001 1000	12
19	1 0011	0001 1001	13
20	1 0100	0010 0000	14
126	111 1110	0001 0010 0110	7E
127	111 1111	0001 0010 0111	7F
128	1000 0000	0001 0010 1000	80
510	1 1111 1110	0101 0001 0000	1FE
511	1 1111 1111	0101 0001 0001	1FF
512	10 0000 0000	0101 0001 0010	200

BCD 码利用 4 位二进制数来表示一个十进制数字中的每一位。从 0 到 9 的数字的相应的二进制形式用 4 位数来表示。在 BCD 系统中，能够通过 4 位数显示的最大十进制数为 9。 58

通过将十进制数字用相应的 BCD 来代替可以得到十进制数的 BCD 表现形式。为了区分 BCD 系统和二进制系统的不同，将 “BCD” 的符号置于数字部分的右端。十进制数 7863的 BCD 符号如图 3-12 所示。

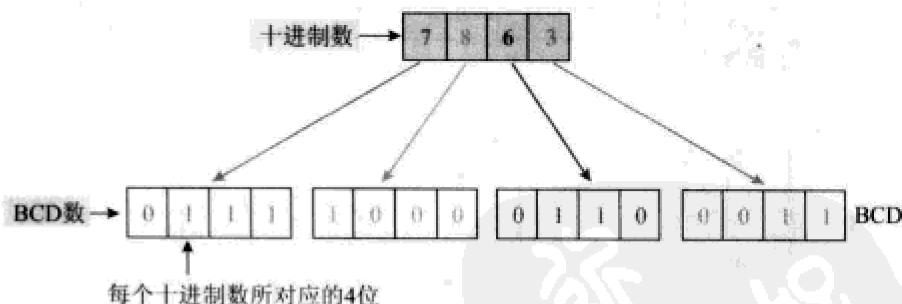


图 3-12 十进制数的 BCD 表现形式

指轮开关是使用 BCD 码的输入设备的一个例子。图 3-13 中显示为单位数 BCD 指轮开关。和指轮开关连接的电路板对每个位都有一个接点，还有一个公共接点。操作员在十进制数0 到9 之间拨号，然后指轮开关输出相应的4 位 BCD 数据。在此例中，拨数字8 将产生 1000 的位输入型式。四位数字指轮开关和单位数指轮开关类似，可以控制总共 16 (4 ×4)个 PLC 输入端。

科学计数器适用于十进制数、二进制数、八进制数和十六进制数之间的相互转换。它价格便宜且使用方便；例如，要将一个以十进制显示的数转换为二进制显示的数，只需要简单地单击一个按键将十进制显示模式改变为二进制显示模式。另外，如图 3-14 所示，大多数 PLC 都包括数字转换功能。如图 3-14a 所示，在输入端需要将 BCD 数字转换为二进制数字。在输出端需要将二进制数字转换为 BCD 数字。注意在图 3-14b 中，十进制转换指令将使源地址中的二进制位型式 N7:23，转换为作为目的地址中相同十进制值的 BCD 位型式 O:20。每当程序被扫描时指令就被执行一次，且此时指令为真。 59

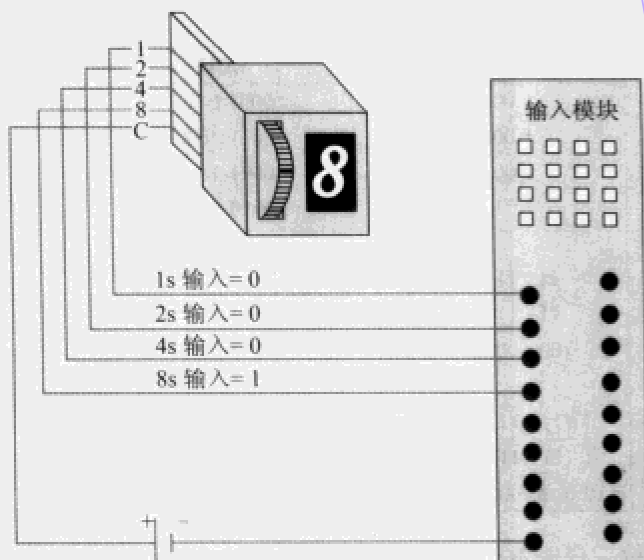
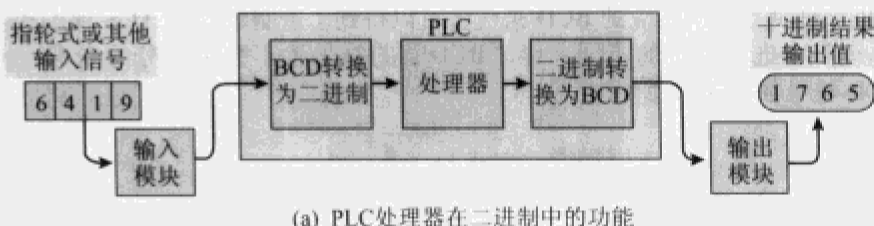
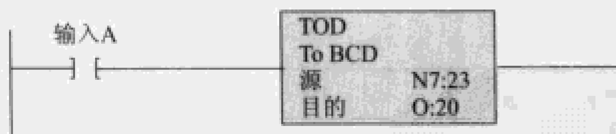


图 3-13　和 PLC 相连的 BCD 指轮式开关



(a) PLC处理器在二进制中的功能



(b) 十进制转换指令

图 3-14　PLC 数字的转换

有些 PLC 允许人为改变数据监视器显示的数据格式。例如，在 Allen-Bradley 控制器中的基数转变功能允许人为将数据格式转变为二进制、八进制、十进制、十六进制或 ASCII。

3.7　格雷码

格雷码是一种特殊的二进制码，它没有使用位加权。换言之，每个位都没有一个确定的权值。通过格雷码可以只改变一个位，就从一个数变为下一个数。在计数器电路中也许容易混乱，但是在编码器电路中是非常标准的。例如，绝对编码器作为位置变送器，它可以利用格雷码来确定角位置。使用格雷码的好处是针对每个“数”（从一个数转变为下一个数）只需要改变一个位。表 3-7 显示了格雷码和相应二进制的比较。在二进制数制中，改变单一的“数”最多需要改变 4 位数字。例如，将二进制数 0111 转变为 1000（十进制数 7 转变为 8）需要改变所有 4 个数字。这种变化增加了在数字电路中出错的可能性。因此，将格雷码作为一种错误最少的编码。由于一次只需改变一个位，格雷码的转变速度比其他码的转变速度要快，如 BCD。

表 3-7 格雷码和相应的二进制数

格雷码	二进制数	格雷码	二进制数	格雷码	二进制数	格雷码	二进制数
0000	0000	0110	0100	1100	1000	1010	1100
0001	0001	0111	0101	1101	1001	1011	1101
0011	0010	0101	0110	1111	1010	1001	1110
0010	0011	0100	0111	1110	1011	1000	1111

格雷码适用于机器人运动、机床和伺服传动系统精确控制的位置编码。图 3-15 显示了利用 4 位格雷码的光学编码器来检测角位置的变化。在此例中，编码器码盘附在转轴上且输出一个数字格雷码信号，此信号用于确定转轴的位置。一组固定的光电二极管用于检测从编码器的径向一列单元的反射光。每个单元将输出一个对应于二进制数 1 或 0 的电压，这取决于光的反射量。因此，码盘上的每一列单元将产生一个不同的 4 位字。

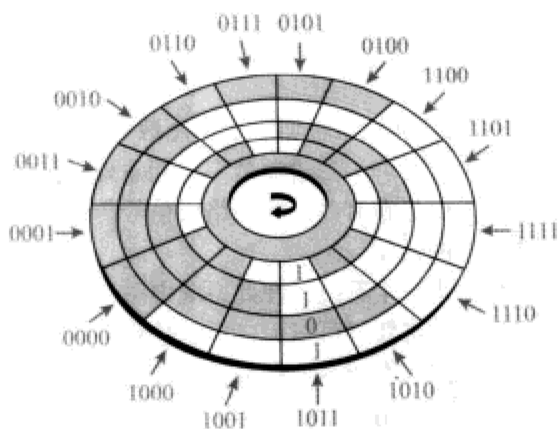


图 3-15 利用格雷码来检测角位置变化的编码器码盘

60

3.8 ASCII 码

ASCII 即美国信息交换标准代码。由于 ASCII 码包括数字和字母，因此它是一种字母数字代码。通过 ASCII 码存取的字符包括 10 个数字；26 个小写字母和 26 个大写字母；大约 25 个特殊符号，其中包括标准打字机里的一些符号，例如，@、#、$、%、* 等。表 3-8 显示了 ASCII 码的部分列表。它用于 PLC 的 CPU 与字母数字键盘及打印机的连接。

表 3-8 ASCII 码的部分列表

字母	7 位 ASCII	字母	7 位 ASCII	字母	7 位 ASCII
A	100 0001	P	101 0000	4	011 0100
B	100 0010	Q	101 0001	5	011 0101
C	100 0011	R	101 0010	6	011 0110
D	100 0100	S	101 0011	7	011 0111
E	100 0101	T	101 0100	8	011 1000
F	100 0110	U	101 0101	9	011 1001
G	100 0111	V	101 0110	空格	010 0000
H	100 1000	W	101 0111	.	010 1110
I	100 1001	X	101 1000	,	010 1100
J	100 1010	Y	101 1001	+	010 1011
K	100 1011	Z	101 1010	-	010 1101
L	100 1100	0	011 0000	#	010 0011
M	100 1101	1	011 0001	(	010 1000
N	100 1110	2	011 0010	%	010 0101
O	100 1111	3	011 0011	=	011 1101

被敲击的计算机键盘上的按键将直接被转换为 ASCII 码，然后进行处理。在键盘上每按下一个键，存储器中就会存储一个 7 bit 或 8 bit 的字，这个字可以用来表示数字字母、函数或表示由于按下特殊键所产生的控制信号数据。ASCII 输入模块能将 ASCII 码输入的外部设备信号转换为 PLC 能够处理的字母数字信息。通过 RS-232 或 RS-422 协议可以实现通信。模块用于传送和接收 ASCII 文件和建立一个操作员接口界面。用户以 BASIC 语言编写程序，随着程序的运行，BASIC 语言和梯形图一起被操作。

3.9 奇偶位

有些 PLC 通信系统使用二进制数字来检验数据传递的准确性。例如，当数据在 PLC 之间传递时，有一个二进制数字可能意外地从 1 变为 0。这种情况的发生是由于瞬变或噪声，或者是由于在有些传输网络中信号传送失败所造成的。奇偶位用于检测在移动一个字时可能出现的错误。

61

奇偶作为一个系统，在此系统中每个传输的数据都包含一个附加的位。这个附加的位称为奇偶位。奇偶位可以为二进制数 0 或 1，这取决于在字符中 1 和 0 数字的个数。

通常使用的奇偶系统有两种：奇数和偶数。奇数奇偶系统表示字符中的位，其中包括奇偶位，为二进制数 1 的总位数为奇数。偶数奇偶系统表示字符中的位，其中包括奇偶位，为二进制数 1 的总位数为偶数。奇数和偶数奇偶位的例子如表 3-9 所示。

62

表 3-9 奇数和偶数奇偶位

字符	偶数奇偶位	奇数奇偶位	字符	偶数奇偶位	奇数奇偶位
0000	0	1	0101	0	1
0001	1	0	0110	0	1
0010	1	0	0111	1	0
0011	0	1	1000	1	0
0100	1	0	1001	0	1

3.10 二进制算术

运算电路单元构成了 CPU 的一部分。数学运算包括加法、减法、乘法和除法。二进制加法运算的规则和十进制加法运算的规则相似。当两个二进制数相加时，只可能出现四种情况：

$$\begin{array}{r}0\\+0\\\hline0\end{array}\qquad\begin{array}{r}1\\+0\\\hline1\end{array}\qquad\begin{array}{r}0\\+1\\\hline1\end{array}\qquad\begin{array}{r}1\\+1\\\hline0\end{array}\ \text{进1}$$

前三种情况非常简单，因为它们和十进制数的加法一样，但是最后一种情况有一点儿不同。在十进制中，1 + 1 = 2。在二进制中，2 被写作 10。因此，在二进制中，1 + 1 = 0，同时向下一个最高有效位的值进 1。当对较大的二进制数进行加法运算时，进位 1 加在高次列上，如下例所示。

十进制	相应的二进制
5 +2 = 7	101 + 10 = 111
10 +3 = 13	1010 + 11 = 1101（进位）
26 +12 = 38	11010 + 1100 = 100110（进位 进位）

在算术函数中，通过减法来结合的两个原始数称为被减数和减数。减法运算所得的结果称为差，表示如下：

```
  A   (被减数)
 -B   (减数)
  C   (差)
```

要实现一个较大二进制数的减法运算，应该对应列的数字减去对应列的数字，当不够减时，需要从相邻的列借位。当从邻列借位时，记住，只有两个数，即，1借给0得到10。

例 1101减去1001。　　1011减去0111。

```
  1101        1011
 -1001       -0111
  0100        0100
```

二进制数也可能为负。由于较小的数被从较大的数中减去，且在结果前加上负号，因此计算过程和十进制数的计算过程是相同的。

例 100减去111。　　11011减去10111。

```
  111         11011
 -100        -10111
 -011        -00100
```


减法运算还有其他一些有用的方式：

第一种补码　第二种补码

使用第一种补码的数进行减法运算的过程如下：

步骤1：将减数变为第一种补码。

步骤2：将两数相加。

步骤3：去掉最后一个进位，且将进位数加在结果的第一位中（循环进位）。

```
十进制          二进制
  10         1010                   1010
 - 6       - 0110   —第一种补码→   +1001
   4          100                  10011
                    循环进位   →      +1
                                     100
```

当在结果的最后有进位时，结果为正。当结果最后没有进位时，结果为负且在结果前加上负号。

例 01101减去11011。

```
    01101
+ Ø 00100   第一种补码
    10001   没有进位，因此采用第一种
            补码且在结果前加上负号：
            -01110
```

在使用第二种补码进行减法运算时，将第二种补码和被减数相加而不是相减。如果进位为1，结果为正；如果进位为0，结果为负且需要给结果加上负号。

例 111 减去101。

```
     111
+ Ø011   第二种补码
    1010   左边第一位数为1表示结果
           为正，因此可以忽略：
           010
```

例 01101减去11011。

```
       01101
+ Ø00101   第二种补码
       10010   没有进位，因此结果为负；所以
               将结果减去1之后，结果变为
               第一种补码形式：
减1            10010-1=10001
第一种补码     -01110
```

二进制数的相乘和十进制数的相乘类似。当两个二进制数相乘时，只可能出现四种情况：

0×0=0 0×1=0 1×0=0 1×1=1

为了实现多位数字的相乘，需要得出局部乘积且将它们相加，如下例所示。

```
十进制乘法          相应的二进制乘法
   5                   101
  ×6                 × 110
  30                   000
                      101
                     101
                     11110
```

二进制数的除法运算过程和十进制数的除法相似，如下例所示。

```
十进制除法          相应的二进制除法
                         111
                    10)1110
                       10
                        11
                        10
      7                  10
   2)14                  10
                         00
```

比较函数的基本功能用于比较两个量的相对大小。PLC 数据的比较指令用于比较存储在两个字（或寄存器）中的数据。有时，在应用中，当出现数据值小于、等于或大于其他数据值的情况时，需要对设备进行控制或设置工作点，如定时器和计数器的值。基本比较指令如下：

A = B（A 等于 B）　A > B（A 大于 B）　A < B（A 小于 B）

65

思考题

1. 将如下各二进制数转换为十进制数：
   a. 10　b. 100　c. 111　d. 1011　e. 1100　f. 10010　g. 10101　h. 11111　i. 11001101　j. 11100011
2. 将如下各十进制数转换为二进制数：
   a. 7　b. 19　c. 28　d. 46　e. 57　f. 86　g. 94　h. 112　i. 148　j. 230
3. 将如下各八进制数转换为十进制数：
   a. 36　b. 104　c. 120　d. 216　e. 360　f. 1516

66

4. 将如下各八进制数转换为二进制数：
   a. 74　b. 130　c. 250　d. 1510　e. 2551　f. 2634
5. 将如下各十六进制数转换为十进制数：
   a. 5A　b. C7　c. 9B5　d. 1A6
6. 将如下各十六进制数转换为二进制数：
   a. 4C　b. E8　c. 6D2　d. 31B
7. 将如下各十进制数转换为 BCD 码：
   a. 146　b. 389　c. 1678　d. 2502
8. 格雷码的最重要的特征是什么？
9. 什么使得二进制系统非常适用于计算机电路？
10. 通过运用于二进制存储单元或寄存器定义如下各术语：
    a. 比特　b. 字节　c. 字　d. 最低有效位　e. 最高有效位

67

11. 针对如下数字系统，列出其基数：
    a. 八进制　b. 十进制　c. 二进制　d. 十六进制
12. 定义术语符号位。
13. 解释第一种补码和第二种补码的不同点。
14. 什么是 ASCII 码？
15. 为什么要使用奇偶位？
16. 对如下二进制数进行加法运算：
    a. 110 + 111　b. 101 + 011　c. 1100 + 1011
17. 对如下二进制数进行减法运算：
    a. 1101 − 101　b. 1001 − 110　c. 10111 − 10010
18. 对如下二进制数进行乘法运算：
    a. 110 × 110　b. 010 × 101　c. 101 × 11
19. 对如下二进制数进行除法运算：
    a. 1010 ÷ 10　b. 1100 ÷ 11　c. 110110 ÷ 10

68

习题

1. 将如下二进制的 PLC 序列码信息转换为十六进制码编程。将每组二进制信息转换为适当的十六进制信息，使之能通过键盘输入到 PLC 内。

a. 00011111　b. 00100101　c. 01001110　d. 00111001

2. 如图 3-16 所示编码器电路用于将十进制数通过键盘转换为二进制编码。当输入如下十进制数时，描述 A－B－C－D 端的输入状态（高/低）。

   a. 按下 2　b. 按下 5　c. 按下 7　d. 按下 8

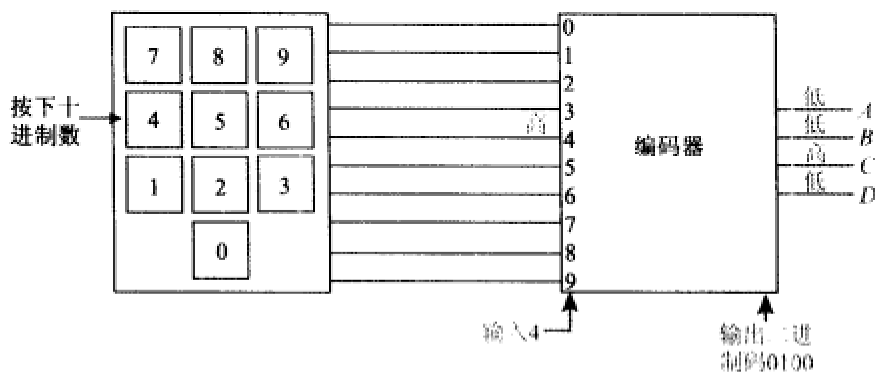


图　3-16

3. 如果一个 16 bit 字或寄存器的位是按照八进制数制系统来编号，从 00 开始，接下来连续的用于表现各位的编号是什么？

4. 用如下各数字码来表达十进制数 18：

   a. 二进制　b. 八进制　c. 十六进制　d. BCD

69

第 4 章　逻辑基础

学习目标：

- ❑ 描述二进制原理及门电路的功能。
- ❑ 画出逻辑符号，构造出真值表并且陈述关于与、或及非功能的布尔方程。
- ❑ 通过布尔表达式构造电路以及针对已知逻辑电路导出布尔方程。
- ❑ 将继电器梯形原理图转化为梯形图程序。
- ❑ 基于逻辑门功能初步编写程序。
- ❑ 编写能演示逻辑运算的指令。

本章综述了数字逻辑门并且说明了怎样通过 PLC 来重新实现这种控制方式。布尔代数是一种描写数字门图表的速记法，本书对它进行了简单的讨论。在少数的情况下，一些手持编程器上还有数字逻辑键，例如“与”、“或”和“非”，可以利用布尔表达式来编写程序。

4.1　二进制的概念

如同所有的数字型设备一样，PLC 也是利用二进制原理进行工作的。二进制原理引用的思想是：许多事物都可以描述为存在于两种状态当中的一种。这种状态可以被定义为“高”或“低”，“开”或“关”，“是”或“非”以及“1”或“0”。例如，一盏灯能够处于开或关的状态，一个开关可以导通或关断，以及一台电机可以处于运行或停止状态。

这两种状态的二元概念能够作为门电路运用当中判定的基础。门电路是一种拥有一个或多个输入端口的设备。通过这些端口，门电路能进行逻辑判定并且在它的一个输出端口产生结果。图 4-1 和图 4-2 给出了两种描述逻辑门电路怎样做出判定的例子。

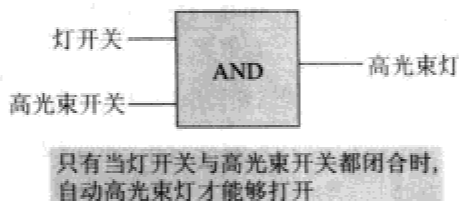


图 4-1　逻辑“与”

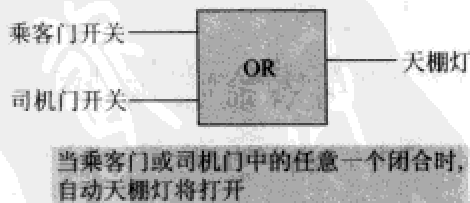


图 4-2　逻辑“或”

当在行为发生之前就需要考虑一个或者多个不同的因素时，逻辑推理就能够做出判定。当需要一个设备运作，那么某些与之有关的条件必须已经满足，这是 PLC 运行的基础。

4.2　与、或、非函数

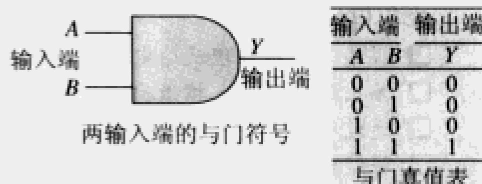
数字型设备完成操作可表示为三种基本逻辑功能，即：“与”、“或”和“非”。每种功能都有一种决定结果的准则以及一个代表操作的符号。为了方便讨论，我们把结果或输出称为 Y，将输入信号称为 A、B、C 等。另外，二进制数“1”表示一种信号的存在或者一些事件的发生，而二进制数“0”则表示信号不存在或者事件没有发生。

4.2.1 与函数

在图4-3中的符号称为与门。与门是一种包含两个或者更多输入端口和一个输出端口的元器件。只有当所有的输入为1时，与门的输出才为1。图4-3中的真值表显示了各种输入组合条件下的输出结果。

71
~
72

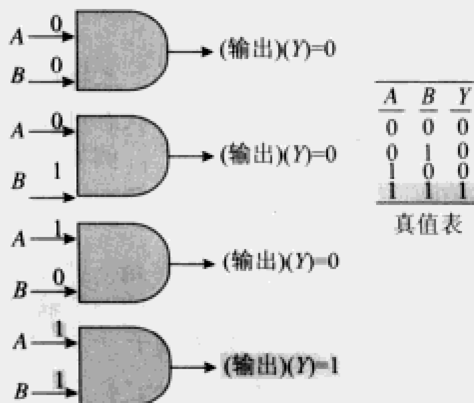
图4-4和图4-5显示了与门函数的实际应用。当控制开关A和B都闭合时，输出Y或者指示灯变为工作状态——它转为接通状态。如果我们把工作状态定义为逻辑1且把非工作状态定义为逻辑0，那么就能形成反映与功能的真值表。当用1来描述工作状态且用0来描述非工作状态的时候，我们使用了正逻辑。与门的工作状态就如同一个串联电路，只有当输入电压出现在它的每一个输入端的时候，它才能产生一个输出电压。



输入端		输出端
A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

与门真值表

图4-3 与门

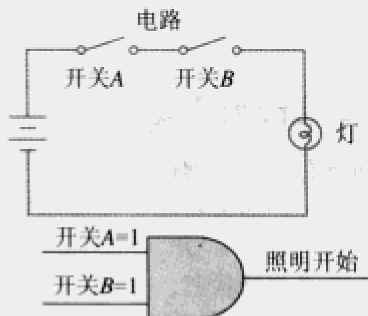


A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

真值表

基本规则：如果所有输入端为1，则输出端将为1。如果任意一个输入端为0，则输出端为0

图4-4 与门函数的应用——例1



开关A		开关B		灯
断开	(0)	断开	(0)	关(0)
断开	(0)	闭合	(1)	关(0)
闭合	(1)	断开	(0)	关(0)
闭合	(1)	闭合	(1)	开(1)

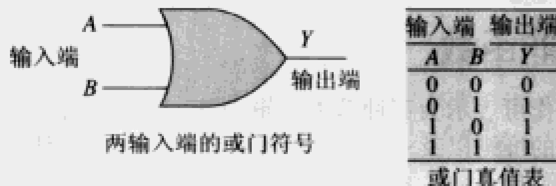
真值表

只有当开关A和开关B都闭合时，灯才会发亮

图4-5 与门函数的应用——例2

4.2.2 或函数

在图4-6中所示的符号称为或门。一个或门可以有很多输入，但只有一个输出。如果或门有一个或者多个输入为1，那么它的输出就为1。图4-6中的真值表显示了各种输入组合条件下的输出结果。

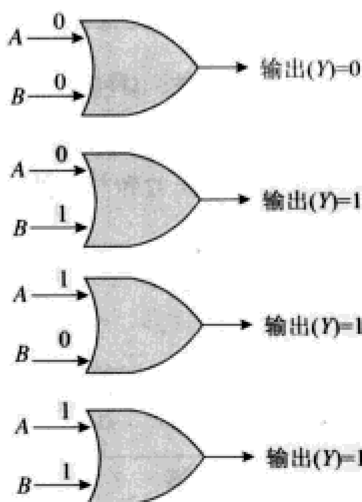


输入端		输出端
A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

或门真值表

图4-6 或门

图 4-7 和图 4-8 显示了或门函数的实际应用。或门本质上就如同一个并联电路，当输入电压出现在它的任意一个输入端的时候，它就产生一个输出电压。

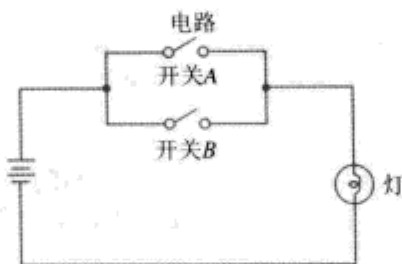


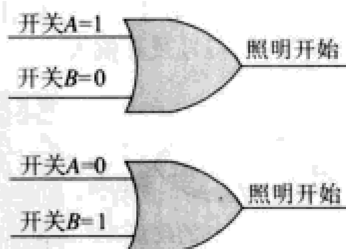
输入端		输出端
A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

真值表

基本规则：如果有一个或多个输入端为1，则输出端为1。如果所有输入端为0，则输出端为0

图 4-7　或门函数的应用——例 1





开关A		开关B		灯	
断开	(0)	断开	(0)	关	(0)
断开	(0)	闭合	(1)	开	(1)
闭合	(1)	断开	(0)	开	(1)
闭合	(1)	闭合	(1)	开	(1)

真值表

如果开关A或开关B闭合，灯将发亮

图 4-8　或门函数的应用——例 2

4.2.3　非函数

在图 4-9 中所示的符号能实现非功能。跟与函数、或函数不同的是，非门只有一个输入端。当非门的输入为 0 时，它的输出为 1；输入为 1 时，非门的输出为 0。非运算的结果通常是将输入信号取反，因此，非门也被称为反相器。我们常常在字母的顶上加上一条横线来表达非函数，表示是一个反相输出。在反相器输出端的小圆称为状态指示符，它表示逻辑函数已经取反。 [73]

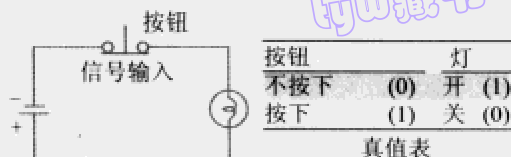
图 4-10 显示了一个关于非函数实际应用的例子。在这个例子中，输出端和一个常闭按钮串联。当没有按下按钮时，输出端为导通状态；当按下按钮时，输出端为断开状态。 [74]



A	NOT A
0	1
1	0

非门真值表

图4-9　非函数的符号



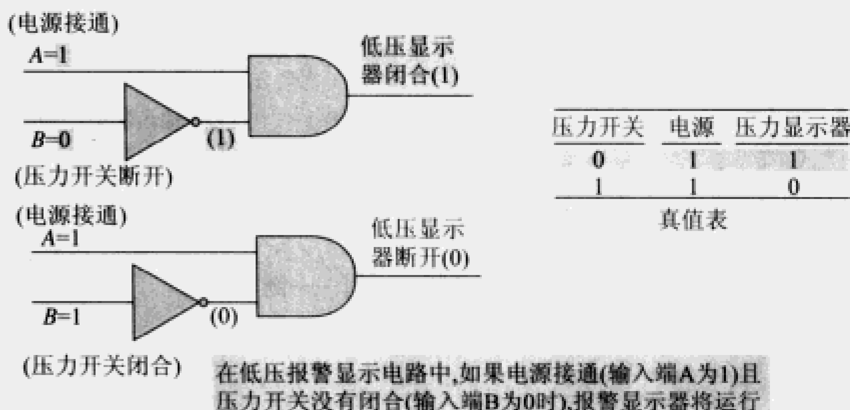
按钮		灯	
不按下	(0)	开	(1)
按下	(1)	关	(0)

真值表

如果按钮没有按下,灯将发亮

图4-10　非门函数的应用——例1

非函数通常是同与门和或门结合起来使用。图4-11显示了非函数和与门的一个输入端相连。



压力开关	电源	压力显示器
0	1	1
1	1	0

真值表

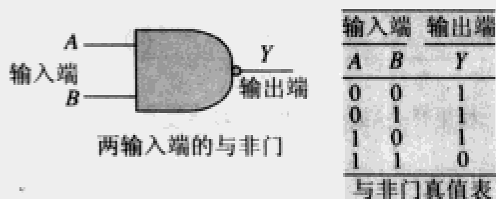
在低压报警显示电路中,如果电源接通(输入端A为1)且压力开关没有闭合(输入端B为0时),报警显示器将运行

图4-11　非函数的应用——例2

把符号非置于与门的输出端将使正常的输出结果反相。输出反相的与门称为与非门。图4-12中显示了与非门的符号及真值表。我们通常在集成电路的逻辑数组中使用与非函数，以及在PLC中用它来解决复杂的逻辑关系。

75

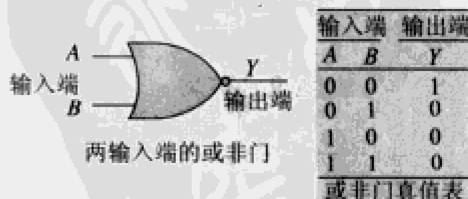
按照同样的将正常输出结果取反运用的规则，如果把符号非置于或门的输出端也会使输出结果反相，此函数称为或非门。图4-13显示了或非门的符号及真值表。



输入端		输出端
A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

与非门真值表

图4-12　与非门电路符号及其真值表



输入端		输出端
A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

或非门真值表

图4-13　或非门电路符号及其真值表

4.2.4　异或函数

异或函数（XOR）（如图4-14所示）是一种常用的组合门电路。只有当两个输入端中的一个输入端为高电平时，此电路的输出才为高电平。而当两个输入端都为高电平或低电平时，输出信号为低电平。异或门通常用于两个二进制数的比较。

输入端		输出端
A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

异或门真值表

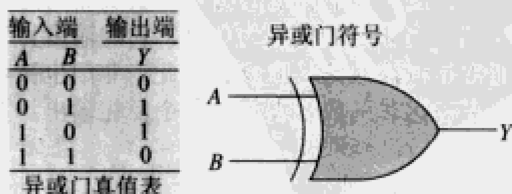


图4-14　异或门符号及其真值表

4.3 布尔代数

我们把关于二进制数系统及逻辑关系的数学研究称作布尔代数。研究布尔代数的目的是为了给描写复杂的组合逻辑语句提供一个简便的书写方法。关于布尔代数的应用有许多，我们可以用它来解决 PLC 当中的编程问题，事实上在一些 PLC 当中，可以直接利用布尔指令来编写程序（如表 4-1 所示）。跟继电器式梯形逻辑（RLL）相比，布尔逻辑更适合大众。因为每个人都知道与、或和非的含义，而对于梯形逻辑来说，除了电气工程师和 PLC 程序员比较熟悉以外，其他人对它并不熟悉。

表 4-1

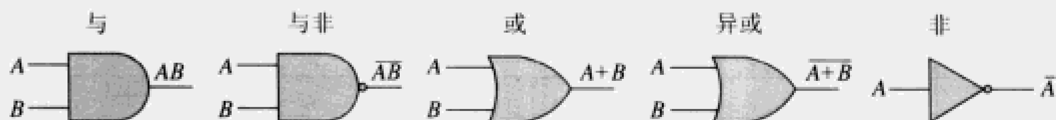
布尔指令及功能	图形符号
STR——LD 用一个常开触点开始一个新的梯级或梯级的一个分支	—┤├—
STR NOT——LD NOT 用一个常闭触电开始一个新的梯级或梯级的一个分支	—┤/├—
OR 在一个梯级中用一个常开触点并联另一个触点	
OR NOT 在一个梯级中用一个常闭触点并联另一个触点	
AND 在一个梯级中的一个常开触点和另一个触点相串联	┤├┤├
AND NOT 在一个梯级中的一个常闭触点和另一个触点相串联	┤├┤/├
AND STR——AND LD 将梯级中电路块与前面的电路串联	
OR STR——OR LOAD 将梯级中电路块与其他电路并联	
OUT 将梯级状态（ON/OFF）和离散输出状态（ON/OFF）反映到指定图像寄存器指示位和寄存单元中	—(OUT) —◯
OR OUT 将梯级状态和离散输出状态（ON/OFF）反映到图像寄存器。参考同样的离散指示位，多个 OR OUT 指令也能一起用于程序中	—(OR OUT)
OUT NOT 反映梯级状态和将输出从 OFF 转换为 ON 执行状态；或将输出从 ON 转换为 OFF 执行状态	—⊘

图 4-15 概括了涉及与、或和非基本函数的布尔代数的基本运算。输入端由大写字母 A、B、C 等表示，输出端由字母 Y 表示。乘法标志（×）或圆点（·）代表与运算，加法标志（+）代表或运算，空心圆中含有一个加号的符号（⊕）代表异或运算，在一个字母上加一横线（$\overline{A}$）表示非运算。

逻辑符号	逻辑状态	布尔方程	布尔标示方法
A, B → Y	如果A和B为1， 则Y为1	$Y=A\cdot B$ 或 $Y=AB$	符号　含义 ·　与 +　或 [illegible]　非 [illegible]　反 =　结果
A, B → Y	如果A或B为1， 则Y为1	$Y=A+B$	
A → Y	如果A为0则Y为1， 如果A为1则Y为0	$Y=\overline{A}$	

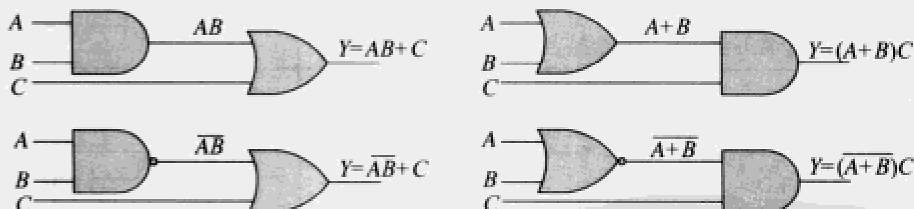
图 4-15　布尔代数相关的与、或和非指令

我们可以利用布尔代数来设计数字系统，通过布尔方程来表达电路的函数。怎样用基本与、或和非函数来列布尔方程，对此可以参看图 4-16 和图 4-17 的两个例子。



基本逻辑门可以完成简单的逻辑功能。如图所示，每个逻辑功能都可以通过布尔表达式来表达

图 4-16　布尔方程——例 1



任何控制功能的组合都可以表示为如图中所示

图 4-17　布尔方程——例 2

76
~
77

了解了用简单的布尔方程来描写复杂的逻辑状态之后，当创建 PLC 控制程序时会很有帮助。布尔代数的某些定律与普通代数不同。以下三条基本定律说明了布尔代数与普通代数之间细微的比较，以及两者之间的一个主要区别。

交换律

$A+B=B+A$

$A\cdot B=B\cdot A$

结合律

$(A+B)+C=A+(B+C)$

$(A\cdot B)\cdot C=A\cdot(B\cdot C)$

分配律

$A\cdot(B+C)=(A\cdot B)+(A\cdot C)$

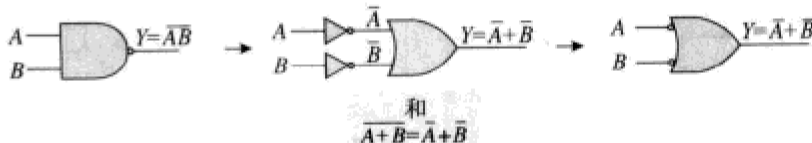
$A+(B\cdot C)=(A+B)\cdot(A+C)$

此定律只在布尔代数中有效

德摩根定律是布尔代数中一个重要的结论。任何一个逻辑函数可以通过与门电路与反相器或或门电路与反相器来执行（如图 4-18 所示）。

根据德摩根定律

$\overline{AB}=\bar{A}+\bar{B}$



和

$\overline{A+B}=\bar{A}+\bar{B}$

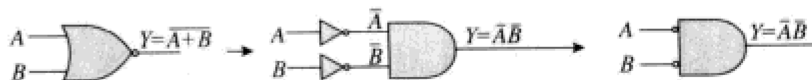
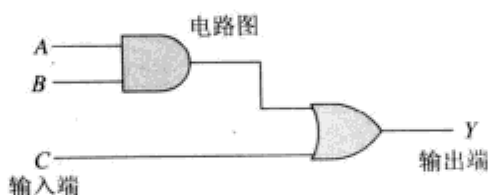


图 4-18 德摩根定律

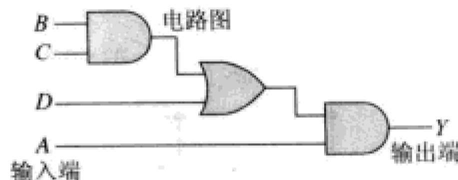
4.4 通过逻辑表达式设计电路

随着逻辑电路变得越来越复杂，对这些电路用布尔代数表达的需求也越来越多。一个简单的逻辑门电路操作起来很直接。然而将大量的这种门电路组合起来的时候，组合电路的输入信号产生一个什么样的输出信号将变得很难确定。图 4-19 和图 4-20 说明了通过布尔表达式来构造电路的方法。



布尔表达式：$Y=AB+C$
所需的门：（通过检查）
1个　将A和B作为输入端的与门
1个　将C和前一个与门的输出端作为输入端的或门

图 4-19 根据布尔方程构造电路——例 1

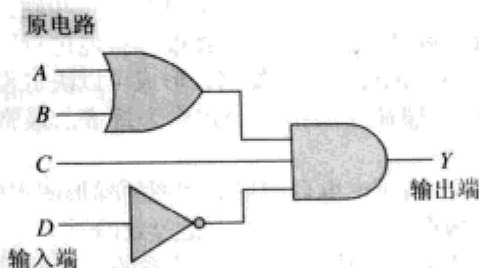


布尔表达式：$Y=A(BC+D)$
所需的门：（通过检查）
1个　将B和C作为输入端C的与门
1个　将$B\cdot C$和D作为输入端的或门
1个　将A和前一个或门的输出作为输入端的与门

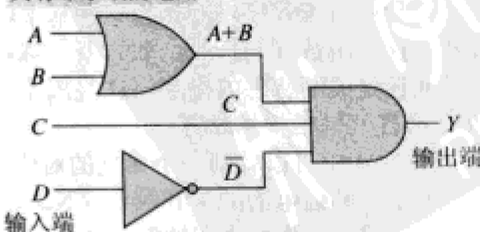
图 4-20 根据布尔方程构造电路——例 2

4.5 通过已知电路构造布尔方程式

图 4-21 和图 4-22 说明了怎样通过已知电路来列出布尔方程。



具有布尔项的电路



最终表达式：$Y=C\bar{D}(A+B)$

图 4-21 根据已知电路列出布尔方程——例 1

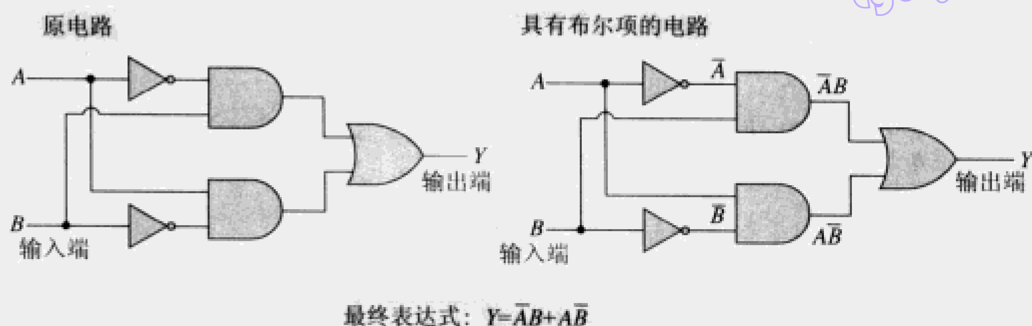


图4-22　根据已知电路列出布尔方程——例2

4.6　针对可编程序逻辑的硬接线逻辑

关于逻辑控制功能的术语*硬接线逻辑*是由器件之间的互联方式来确定的。硬接线逻辑可以通过继电器和继电器梯形原理图来实现。继电器梯形原理图在工业领域被广泛地使用和学习。图4-23显示了带有指示灯的电动机停/启控制台的继电器梯形原理图。控制电路画于两条垂直的馈电线路之间。所有安置在这两条线路之间的元件被称为*梯级*或*支路*，且连接在两根动力线路之间，其形状就像是*梯子的横档*——因此得名为*梯形图程序*。

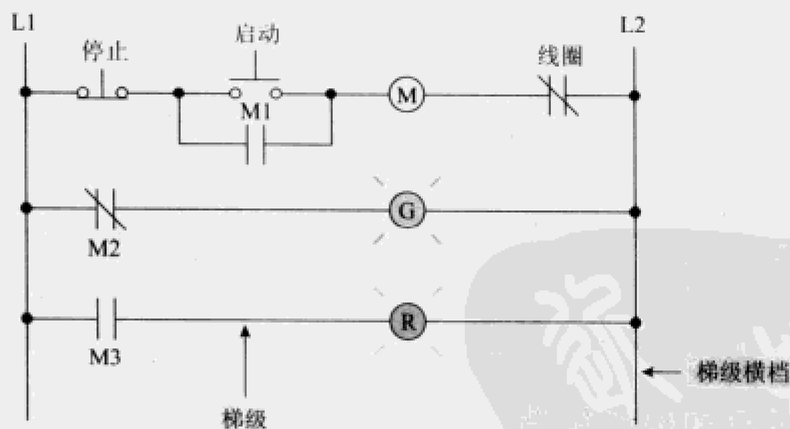


图4-23　继电器式梯形原理图

硬接线逻辑是固定的，只有通过改变设备的连接方式才能改变其逻辑关系。和硬接线逻辑相比较，可编程序控制是建立在基本的逻辑函数之上的，它们能够很容易地进行编程及改变。这些函数（与、或、非）既可以单独调用，也可以组合使用，形成可以决定设备处于开通还是关断状态的指令。这些用来向PLC传递命令的指令形式称为*语言*。最常用的PLC语言就是*逻辑梯形图*。

从图4-24中可以看到一个典型的对应于图4-23所示继电器梯形原理图的梯形图程序。其使用的指令和继电器的常开触点（NO）、常闭触点（NC）以及线圈是等效的。

就继电器式梯形原理图的符号来说，触点符号体系是表现控制逻辑关系的一种简单方法。一个梯级就是一个需要控制输出的触点。一些PLC允许一个梯级有多个输出。因此一个完整的梯形图程序由几个具有各自控制输出的梯级组成。

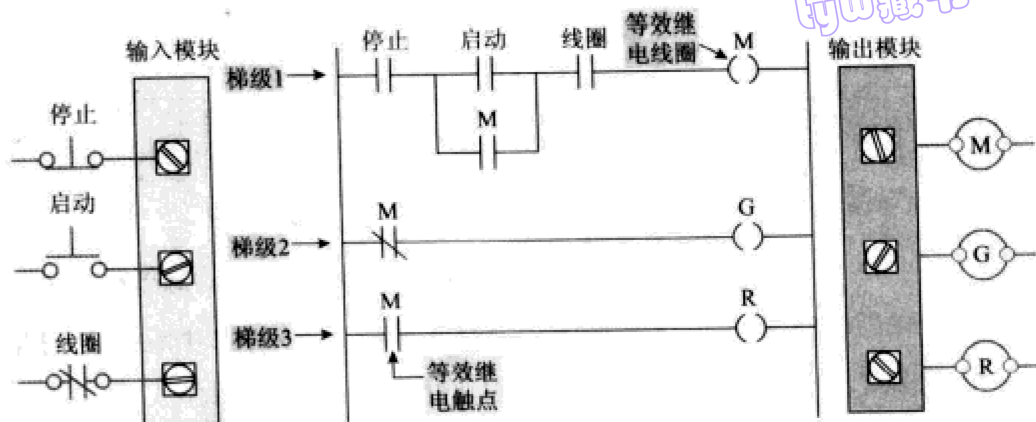
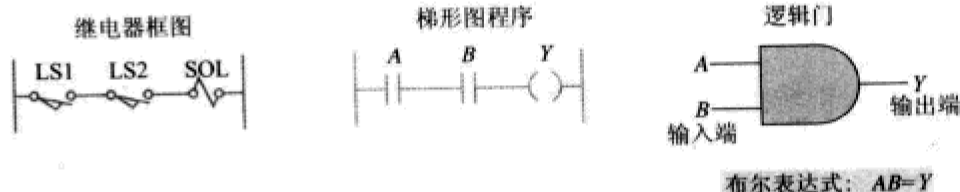


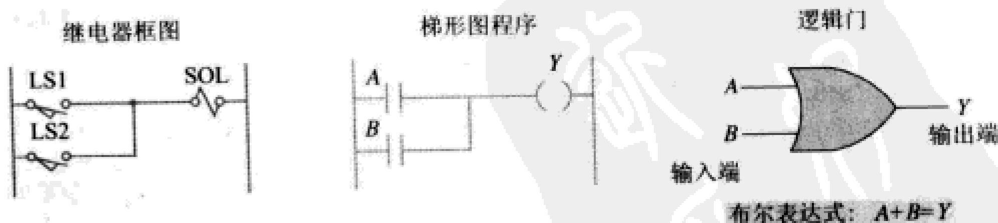
图 4-24 逻辑梯形图程序

由于 PLC 使用的是梯形图程序，因此把任何已存在的继电器逻辑图转换为可编程序逻辑图非常容易。每个梯级都是输入条件（符号体系）从左到右的连接，在其最右端是输出符号。表示输入信号的符号通过串联、并联或一些串、并联组合的连接方式来获得所期望的逻辑关系。以下几组例子说明了继电器梯形原理图、PLC 梯形图程序以及等效逻辑门电路之间的关系。（见例 4-1 至例 4-9）。

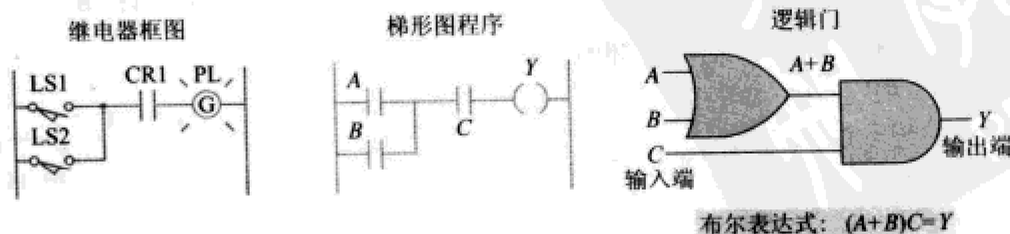
81 ↓ 82



例 4-1 两个限位开关串联对螺线管操作阀实行控制

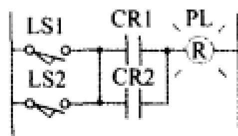


例 4-2 两个限位开关并联对螺线管操作阀实行控制

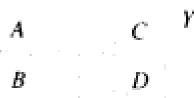


例 4-3 两个限位开关并联后串联一个继电器触点对指示灯实行控制

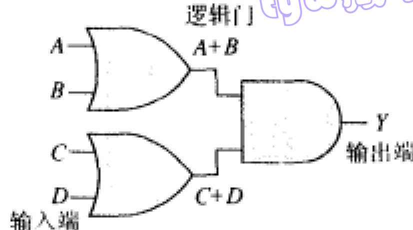
继电器框图



梯形图程序



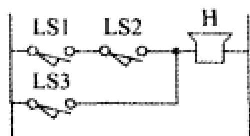
逻辑门



布尔表达式：$(A+B)(C+D)=Y$

例4-4　两个限位开关并联之后同另外两个触点（触点之间为并联）串联对指示灯实行控制

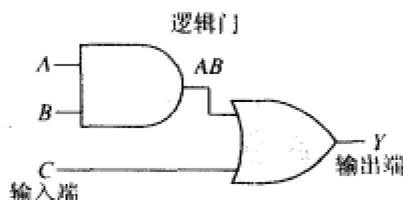
继电器框图



梯形图程序



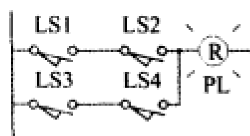
逻辑门



布尔表达式：$(AB)+C=Y$

例4-5　两个限位开关串联之后同第三个限位开关并联对报警喇叭实行控制

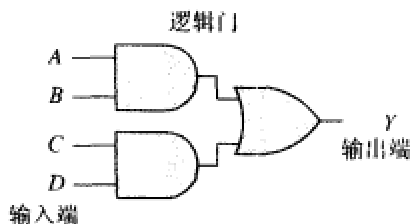
继电器框图



梯形图程序



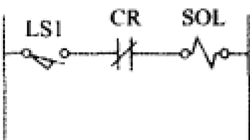
逻辑门



布尔表达式：$(AB)+(CD)=Y$

例4-6　两个限位开关串联之后同另外两个开关（开关之间为串联）并联对指示灯实行控制

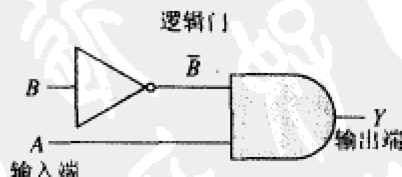
继电器框图



梯形图程序



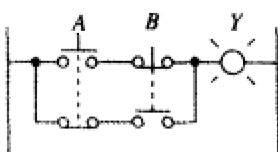
逻辑门



布尔表达式：$A\bar{B}=Y$

例4-7　一个限位开关与一个数控继电器触点串联对螺线管操作阀实行控制

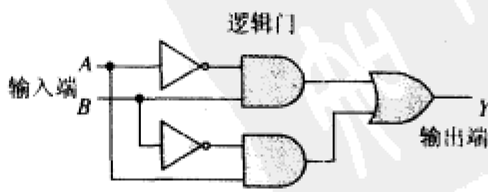
继电器框图



梯形图程序



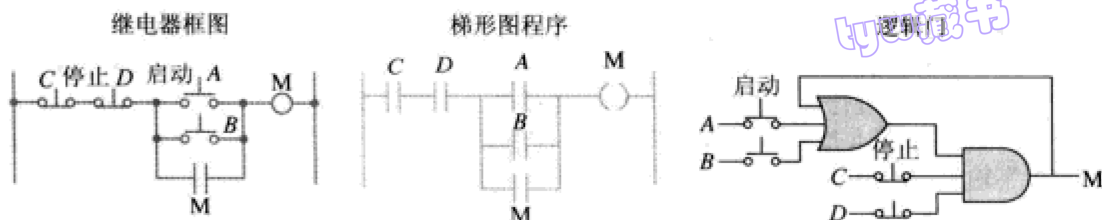
逻辑门



布尔表达式：$\bar{A}B+A\bar{B}=Y$

$A\oplus B=Y$

例4-8　异或电路。只有当按钮 *A* 或 *B* 而不是同时被按下时，电路的输出才为接通状态



例 4-9　具有两个启动和停止按钮的电机控制电路。当按下启动按钮，电机运转。由于自锁，在松开启动按钮之后电机仍继续运转。当按下停止按钮时，电机停止运行

83
~
85

4.7　字级编程逻辑指令

大多数的 PLC 都提供字级逻辑指令作为其指令系统的一部分。表 4-2 显示了针对不同情况选择正确的逻辑指令。

表 4-2　逻辑指令的选择

如果你想确认什么时候	使用这个指令
分别位于两个不同字中的匹配位都为接通状态	与
分别位于两个不同字中的其中一个或两个匹配位为接通状态	或
分别位于两个不同字中的一个或另一个匹配位为接通状态	异或
将字当中的位的状态取反	非

图 4-25 说明了运用与指令在两个源地址中逐位执行与运算。当输入设备 *A* 状态为真时，处理器将按照指令要求将 B3:5 和 B3:7 逐位进行与运算并且将运算结果存储在目标单元 B3:10。

图 4-26 说明了或指令的运算。将信号源 *A* 的数据按位对位的方式与信号源 *B* 中的数据进行或操作且将结果储存在目标单元中。信号源 *A* 的地址为 B3:1，信号源 *B* 的地址为 B3:2，目标单元地址为 B3:20。这个命令可能是有条件编程，即由输入的命令预测结果；也可能是无条件编程，即没有任何输入的命令来预测。

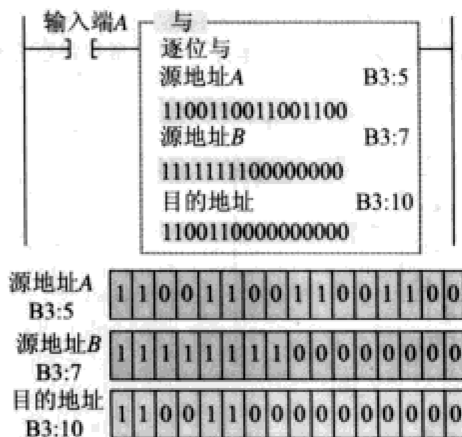


图 4-25　与指令(目标位为逻辑与运算的结果)

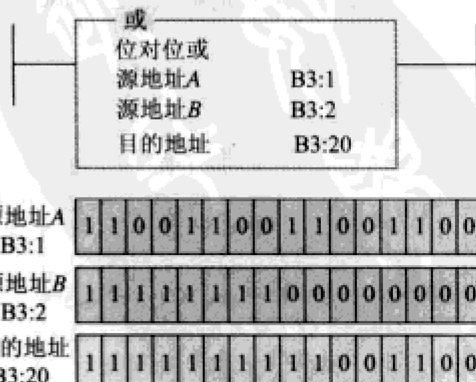


图 4-26　或指令(目标位为逻辑或运算的结果)

图 4-27 说明了异或指令的运算。在这例子中，输入地址 I:1.0 中的数据按照位对位的方式与输入地址 I:3.0 中的数据进行比较。任何不匹配的数据都会在字 O:4.0 中的相应位

产生一个激励。如你所见，当信号 A 和信号 B 不相同时，在其相应的每个目标位单元的值为1；当信号 A 和信号 B 相同时，在其相应的每个目标位单元的值为0。在实际输入信号需要与其期望状态进行比较的情况下，如限位开关，通常需要异或运算来进行诊断。

86

图4-28说明了非指令的运算。非指令的作用是将处于源字当中的位进行取反，然后从目标字输出。B3:10中的位状态就是将B3:9中的位取反所得到的结果。

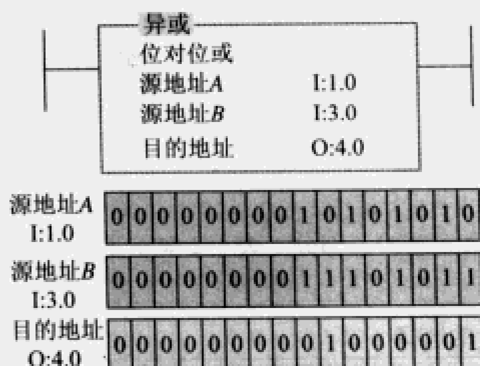


图4-27　异或指令

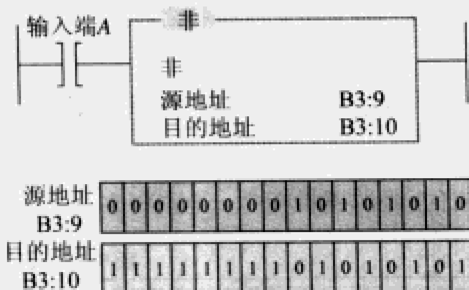


图4-28　非指令（目标位为逻辑非运算的结果）

对于32位PLC，如Allen-Bradley逻辑控制器，其源和目标值可以为一位整数、双位整数或四位整数值。

87

思考题

1. 解释二进制原理。
2. 电子门的作用是什么？
3. 按照下列各个要求，分别画出逻辑符号，建立真值表并且列出状态方程：
   a. 两输入的与门　b. 非函数　c. 三输入的或门　d. 异或函数
4. 利用梯形图程序表述下列各方程：
   a. $Y=(A+B)CD$　　b. $Y=A\overline{B}C+\overline{D}+E$
   c. $Y=[(\overline{A}+\overline{B})C]+DE$　　d. $Y=(\overline{A}B\overline{C})+(D\overline{E}F)$
5. 根据如图4-29所示的两个继电器梯形图写出梯形图程序，画出逻辑门电路且列出布尔状态方程。

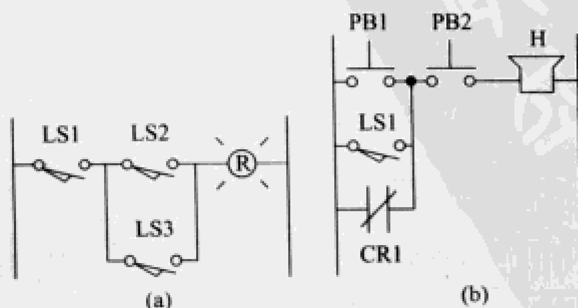


图　4-29

6. 按照以下给出的布尔表达式，利用与门、或门和非门电构造逻辑门电路：
   a. $Y=ABC+D$　b. $Y=AB+CD$　c. $Y=(A+B)(\overline{C}+D)$
   d. $Y=\overline{A}(B+CD)$　e. $Y=\overline{A}B+C$　f. $Y=(ABC+D)(E\overline{F})$

7. 叙述你需要使用的逻辑指令：
   a. 确定何时在两个不同字当中的一个或两个匹配位为接通状态
   b. 将一个字中位的状态取反
   c. 确定何时在两个不同字当中的匹配位都为接通状态
   d. 确定何时在两个不同字当中的一个或另一个匹配位为接通状态

88

习题

1. 当满足以下电路的所有要求时，需要使一个指示灯亮：
   - ❑ 电路当中四个压力开关必须闭合
   - ❑ 电路当中至少三分之二的限位开关必须闭合
   - ❑ 复位开关必须断开

   用 AND、OR 和 NOT 门设计能解决上述问题的逻辑电路。
2. 根据图 4-30（a 到 f）中的逻辑门电路写出布尔方程。

89

3. 利用图 4-31 中的模拟逻辑电路来匹配以下情况：
   (a) 为了置得一辆轿车，你必须具备 10 000.00 美元和一个折合价，否则兑现为另付 2 000 美元。
   (b) 两个负责管理的代表和两个来自工会的代表必须出席仲裁会议。如果任意一个组的人没有出席，那么这个会议就取消。
   (c) 为了获得课程学分，你必须注册并且至少通过一门主要课程考试。
   (d) 一对 K 或者一对 A 都可以赢得这手牌。
   (e) 要想成为一名合格的参与者，你必须至少出席会议期间任一天的早会或者下午会当中的任意一个会议。

90

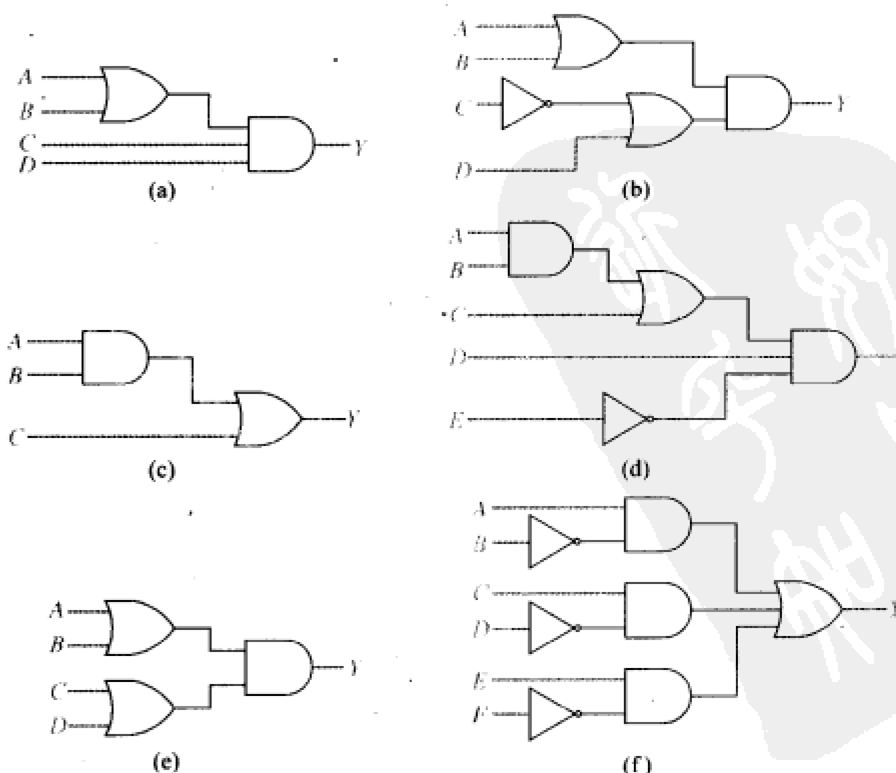


图　4-30

逻辑电路

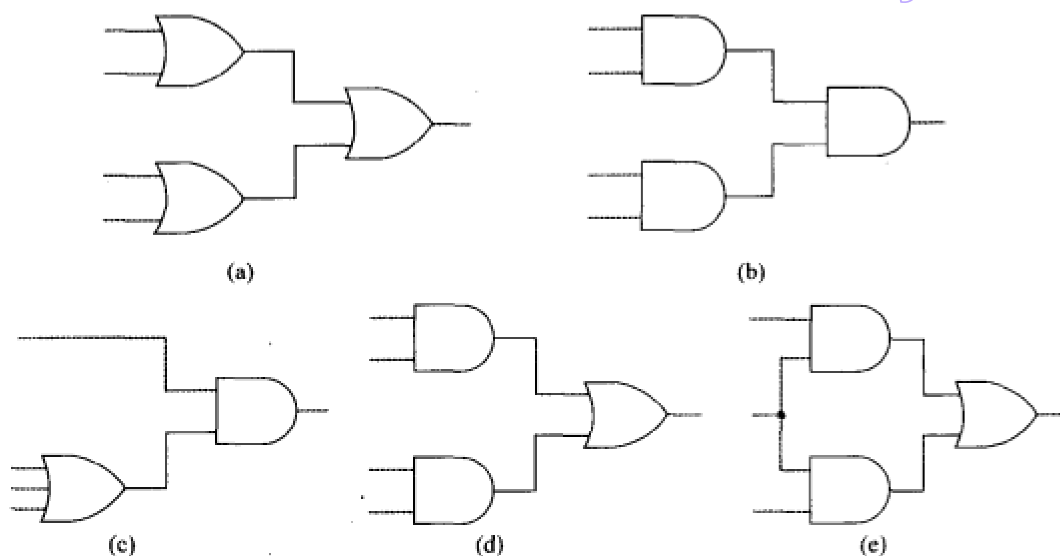


图 4-31

4. 当图4-32中逻辑电路的输出 Y 处于逻辑高电平或1状态时，其作用是使一个警铃工作。对其16种可能的输入状态画出显示输出结果的真值表。

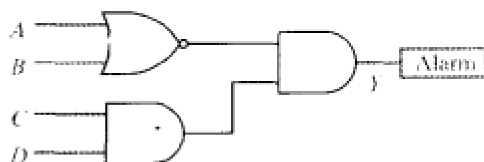


图 4-32

5. 通过下列逻辑运算，数据将以什么值储存在图4-33中的目的地址中？
a. 与运算 b. 或运算 c. 异或运算

源地址 A	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0
源地址 B	0	0	0	0	0	0	0	0	1	1	1	0	1	0	1	1
目的地址																

图 4-33

6. 只有满足以下所有条件风扇才能运行，写出控制系统的布尔方程，画出其门电路逻辑图和标准PLC梯形图。
- ❑ 输入端 A 断开
- ❑ 输入端 B 闭合或输入端 C 闭合，或者 B 和 C 都闭合
- ❑ 输入端 D 和 E 都闭合
- ❑ 输入端 F、G 或 H 中的一个或多个闭合

第 5 章　PLC 编程基础

学习目标：

- ❑ 定义和识别 PLC 的存储映像表功能。
- ❑ 描述输入/输出映像表文件和数据文件类型。
- ❑ 描述 PLC 程序扫描次序。
- ❑ 理解梯形图语言、布尔语言和功能图编程语言如何与 PLC 交换信息。
- ❑ 定义和识别内部继电器指令功能。
- ❑ 识别 PLC 的常见操作模式。
- ❑ 编写并输入梯形图程序。

每个输入、输出 PLC 的模块终端通过唯一的地址来确定。在 PLC 中，任一输入的内部标签都是一触点。类似的，在大多数情况下，所有 PLC 输出的内部标签是一线圈。本章说明了如何使用这些继电器/线圈的功能，对一个电路设计 PLC 程序。本章仅涉及完成类似于继电器功能的基本指令设置，同时还有关于 PLC 程序扫描周期及扫描时间方面的知识。

5.1　处理器存储器结构

处理器存储器结构涉及如何确定 PLC 中存储器的使用状况，并非所有的 PLC 制造商都以同样的方式组织 PLC 存储器。虽然他们使用的存储器的结构、术语不完全相同，但其原理是相同的。

图 5-1 为 Allen-Bradley 公司 PLC-2 的存储器结构图，称为存储映像表。每个 PLC 有一个存储映像表，但它们有可能与图 5-1 所示的不同。存储器空间被划分成两大类：用户程序和数据表。独立区域、次序以及区域长度，可以是固定的或者是变化的，这取决于制造商和模块。

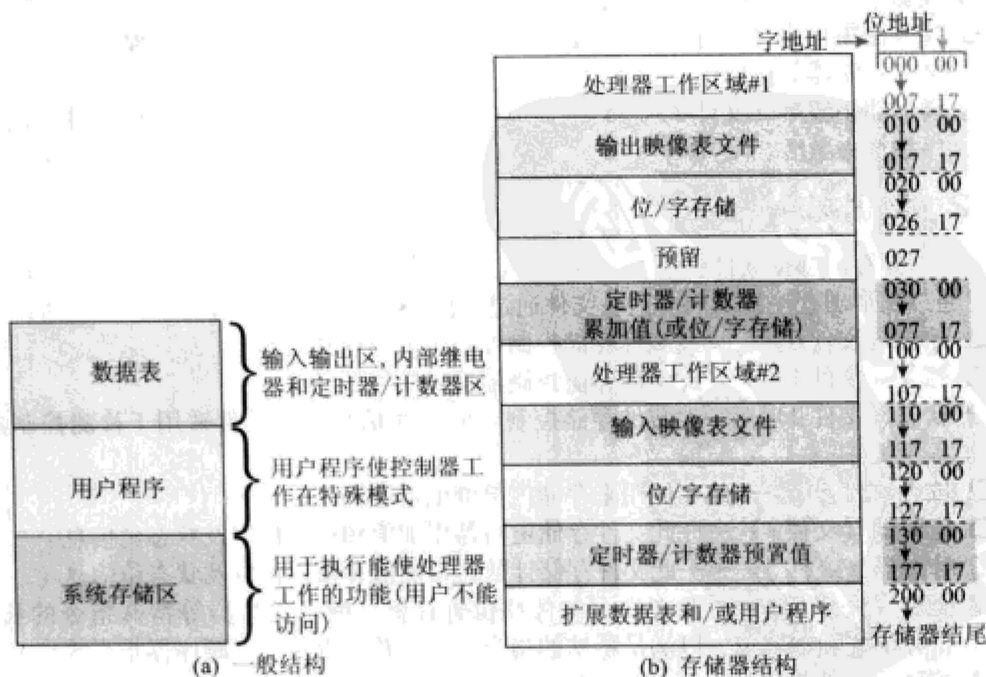


图 5-1　Allen-Bradley 公司 PLC-2 存储映像表

用户程序区是梯形图程序输入和存储的区域。用户程序将占用 PLC 系统存储空间的大部分，它包含控制机器操作的逻辑，这些逻辑包含了以梯形图形式编程的指令。大多数指令需要占用存储器中的一个字。

数据表储存了执行用户程序所需的信息。这些信息包括：输入输出设备状态、定时器和计数器值，数据存储等。数据表中的内容可划分为两类：状态数据和数字或者码字。状态数据是 ON/OFF 类型的信息，用存储在位地址中的 1 和 0 表示。数字或码字信息用一组储存在单个字节或字中的位信息表示。

处理器文件是程序文件和在特殊处理器文件名下创建的数据文件的集合。它包含所有的指令、数据和与用户程序有关的结构配置信息。图 5-2 为Allen-Bradley SLC-500 控制器的典型的程序和数据文件存储器结构。每个文件的内容将在如下部分概述。

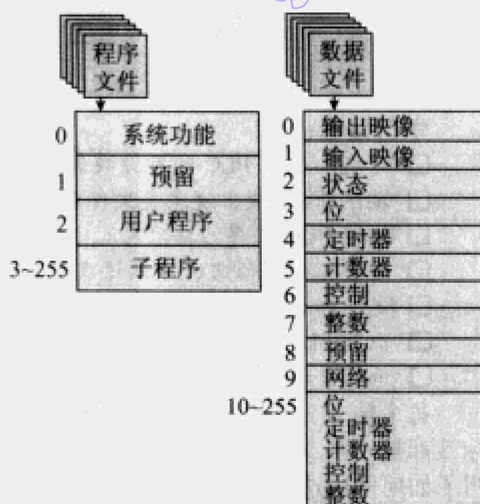


图 5-2　Allen-Bradley 公司 SLC-500 控制器的程序和数据文件存储结构

93~94

5.1.1　程序文件

程序文件是一片处理存储区域，此区域储存梯形图程序，它们包括：

- **系统功能文件**（文件 0）——此文件始终存在，且包含各种相关系统信息和用户编程信息，如处理器类型、I/O 配置、处理器文件名以及口令。
- **预留**（文件 1）——此文件被处理器预留，用户不能访问。
- **梯形图主程序**（文件 2）——此文件始终存在，且包含用户程序结构，此结构定义了控制器如何操作。
- **梯形图子程序**（文件 3～255）——这些文件由用户创建，由主程序文件中的子程序指令来激活。

5.1.2　数据文件

处理器储存的数据文件存储输入输出状态、处理器状态、各种位状态和数据。所有这些信息通过梯形图程序被访问。这些文件通过它们包含的数据类型进行组织，包括：

- **输出**（文件 0）——此文件存储控制器输出的终端状态。
- **输入**（文件 1）——此文件存储控制器输入的终端状态。
- **状态**（文件 2）——此文件存储控制器的操作信息。该文件适用于检测控制器和程序操作。
- **位**（文件 3）——此文件用来存储内部继电器逻辑。
- **定时器**（文件 4）——此文件存储定时器累加值和预设值以及状态位信息。
- **计数器**（文件 5）——此文件存储计数器累加值和预设值以及状态位信息。
- **控制**（文件 6）——此文件存储像移位寄存器、顺序发生器等特殊指令的长度、指针位置和状态位，以满足特殊的指令，如：移位寄存器、顺序器指令等。
- **整数**（文件 7）——此文件用来存储数值和位信息。
- **预定**（文件 8）——此文件用户不能访问。

95

- ❑ **网络通信**（文件 9）——若要安装或使用编号 10～255 的文件，就要采用此文件作网络通信用。
- ❑ **用户定义**（文件 10～255）——这些文件由用户定义，作为位、定时器、计数器、控制以及整数存储。

Allen-Bradley PLC-5 控制器大约有 1000 个程序文件，这些程序文件可能以两种方式建立：第一种为标准梯形图程序，主程序在程序文件 2 中，根据需要程序文件 3 到 999 可以作为子程序；第二种为顺序功能图，根据需要，可以把编号从 2 开始到 999 的文件指定为步或转换。当处理器采用标准梯形逻辑建立时，主程序总是位于程序文件 2 中，程序文件 3 到文件 999 可以作为子程序。每一种情况下，处理器一次只能存储和执行一个程序。

图 5-3 显示了 Allen-Bradley 公司 PLC-5 控制器的典型数据文件存储结构。每个数据文件是由无数个单元组成，每个单元可能是一个，两个，或三个的长度字。定时器、计数器和控制单元占三个字的长度；浮点单元是占两个字长度；而所有其他单元是单个字长度。一个字由 16 位二进制数组成。处理器处理两种不同的数据类型：整数和浮点数。除了浮点文件，所有数据都作为整数处理。在输入、输出数据文件中的所有单元和位地址都用八进制编码，而其他数据文件中的单元和位地址采用十进制编码。典型地址格式如下：

地址范围		单元大小
O:000 O:037	输出映像文件	32
I:000 I:037	输入映像文件	32
S:000 S:031	处理器状态	32
B3:000 B3:999	位文件	1~1000
T4:000 T4:999	定时器文件	1~1000
C5:000 C5:999	计数器文件	1~1000
R6:000 R6:999	控制文件	1~1000
N7:000 N7:999	整数文件	1~1000
F8:000 F8:999	浮点数文件	1~1000
	编号从9到999的文件	1~1000 每个文件

图 5-3　Allen-Bradley 公司 PLC-5 控制器的数据文件存储结构

- ❑ 对于安装在 I/O 机架上面的输入模块和输出模块，输出数据文件和输入数据文件中的地址是潜在的存储区：
  - 输出映像表文件中的地址 O:012/15，代表在导轨 1，第二组 I/O，第 15 位。
  - 输入映像表文件中的地址 I:013/17，代表在导轨 1，第三组 I/O，第 17 位。

96

- ❑ 状态数据文件包含有关处理器状态的信息：
  - 地址 S:015 表示状态文件的第 15 个字的地址。
  - 地址 S:027/09 表示状态文件第 27 个字中第 9 位的地址。
- ❑ 位数据文件存储位状态。当使用内部输出，顺序器，移位指令和逻辑指令时，通常使用它存储：
  - 地址 B3:400 表示位文件的第 400 个字的地址。文件编号（3）必须作为地址的一部分。注意，只有输入、输出和状态数据文件不需要设计文件编号。
  - 字 2 的第 15 位被编址为 B3/47，这是因为位的编号是从文件的起始开始计。记住，位是十进制编号（不是在输入输出中的八进制）。
- ❑ 定时器文件储存定时器状态和定时器数据。定时器单元包含了三个字：控制字，预置字和累加字。定时器控制字的地址是定时器的编号。在文件 4 中的定时器编号从 T4:0 到 T4:999。T4:0 的三个定时器字的地址如下。

控制字：	T4:0
预置字：	T4:0. PRE
累加字：	T4:0. ACC

控制字的使能位地址是 T4:0/EN，定时器定时位地址是 T4:0/TT，完成位地址是 T4:0/DN。

- 计数器文件存储计数器状态和计数器数据。一个计数器单元包含三个字：控制字，预置字和累加字。计数器控制的地址是计数器的编号。在文件 5 中的计数器的编号从 C5:0 开始到 C5:999 结束。计数器 C5:0 的三个计数器字地址如下：

控制字：	C5:0
预置字：	C5:0．PRE
累加字：	C5:0．ACC

在控制字中加计数使能位地址是 C5:0/CU，减计数使能位地址是 C5:0/CD，完成位地址是 C5:0/OV，上溢出位地址是 C5:0/OV，下溢出位地址是 C5:0/UN。

- 控制文件存储控制单元的状态和数据，它用来控制各种文件结构。控制单元包含三个字：控制字，长度字和位置字。控制器的控制字的地址是由被指定的控制编号。在控制文件 6 中的控制单元用 R6:0 作为开始编号，一直到 R6:999。R6:0 中的三个字地址如下：

控制字：	R6:0
长度：	R6:0．LEN
位置：	R6:0．POS

在控制字中有大量的控制位，它们的功能由控制单元中使用的指令决定。

97

- 整数文件储存整数值，其范围从 -32 768 到 32 767。储存的值用十进制数表示，整数单元是一单字（16 位）单元，地址从 N7:000 到 N7:999 的多达 1 000 个整数单元，都能被储存。
  - N7:100 为整数文件的第 100 个字地址。
  - 位地址是十进制，从 0 到 15。例如，在字 15 中的第 12 位被编址为 N7:015/12。
- 浮点数文件单元储存值的范围在 $\pm 1.175\,494\,4 \times 10^{-38}$ 到 $\pm 3.040\,282\,37 \times 10^{+38}$ 之间。浮点单元是双字单元（32 位），多达 1 000 个单元，其地址从 F8:000 到 F8:999，均能被存储。在浮点文件中单个字和位不能被编址。
- 根据需要数据文件 9 通过 999 可能被指定为不同数据类型。当被指定为确定类型时，这个文件被预留给那种类型，而不能用于其他类型。

位文件，整数文件或浮点数文件能被用来储存状态和数据。根据数据的用途可以确定使用哪个文件。如果处理的是状态而不是数据，位文件是首选的。若你使用很大或很小的数且要求十进制小数点，那么浮点文件是首选。然而，浮点数文件类型有个限制，因为它和外部装置或者像计数器、定时器这样的内部指令（只用 16 位字）不能进行很好地连接。在这种情况下，可能需要用整数文件类型。

图 5-4 所示为一个开关通过输入模块与输入映像表文件的连接。当开关闭合时，处理器在输入终端检测到电压，并通过在恰当存储位中储存二进制 1 记录信息。每个连接的输入信号在输入映像表文件中都有与之对应的一位。在 I/O 扫描期间，输入映像表文件发生变化，反映电路中开关的当前状态。若输入处于 ON 态（开关闭合），在映像表中的相应位上被置 1；若输入处于 OFF 态（开关断开），相应位被“清除”或置 0。

输出映像表文件是一个位数组，它控制连接到输出接口电路的数字输出设备的状态。图 5-5 显示了一个灯通过输出模块与输出映像表文件的典型连接。灯的 ON/OFF 状态通过用户程序控制，通过 1（ON）和 0（OFF）来指示。每个连接的输出在输出映像表文件中

都有与之对应的一位。若程序要求一个特定的输出为 ON，在表中它相应的位被置 1；若程序要求是 OFF，在表中它相应的位被置 0。

98

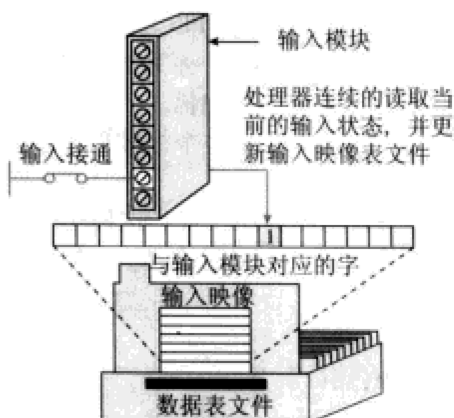


图 5-4 典型输入映像表文件连接

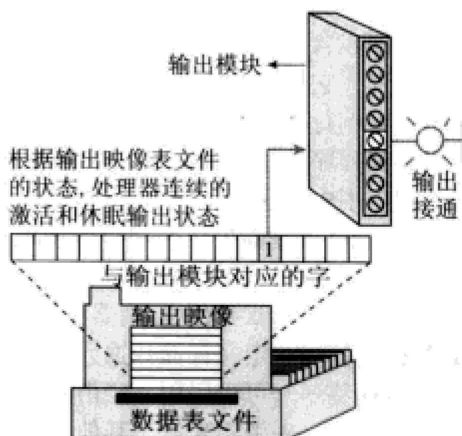


图 5-5 典型输出映像表文件的连接

由于 PLC 的制造商们必须保持和其他控制技术的竞争，PLC 处理器在不断地更新。Allen-Bradley Logix 平台为适应离散、运动及过程控制展示了一个更开放的结构体系。

ControlLogix 处理器提供了一个灵活的存储结构。其他控制器如果用这种方法设计系统，会受到 PLC 数据表的限制。而 ControlLogix 处理器提供数组和用户定义结构，避免了这种限制。这些特征使得数据结构去满足应用的需求，而不是让应用去适应特定的存储结构。

ControlLogix 控制器的项目文件，包含了编程和组态信息。它包含了以下定义的特性：

- ❑ 机架大小/类型；
- ❑ 控制器的槽数目；
- ❑ 说明；
- ❑ 文件路径；
- ❑ 项目名称。

ControlLogix 控制器项目由三个主要部分组成：任务，程序和例程（如图 5-6 所示）。任务对程序进行调度并对分配任务的程序提供优先操作权。有两种任务类型：连续性和周期性。一个连续任务有最低优先权，且总是被周期性任务中断。周期性任务有多达 15 个优先级水平，1 是最高优先级，而 15 是最低优先级。每次执行完连续任务后，会重新启动，但它可能被周期性任务中断。当一个项目被创建时，主任务就被分配为连续任务。被称为具有优先权的多任务操作系统的 ControlLogix 系统有能力中断正在执行的任务，对不同的任务实行开关控制，一旦中断任务执行完返回到最初的任务。在任一时间内只有一个任务在执行，且在一个时间内仅有一个程序正在执行。Logix-5550 控制器可支持 32 个独立任务。

99

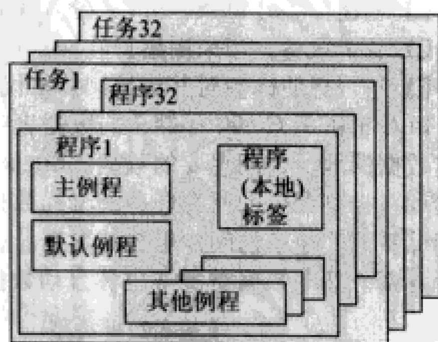


图 5-6 ControlLogix 控制器项目组成

程序组逻辑和数据。每个程序包含一个局部数据区（称为程序标签）、可执行的主程

序、子程序和故障程序。当达到32个程序时，每个程序都有它自己的局部数据和逻辑，且可能被指定为一个任务，在一个任务内程序从头到尾执行。当创建一个工程时，主程序已被定义为主任务。

例行程序包含可执行代码的逻辑。一个例行程序类似于建立在PLC-5或SLC处理器中的程序文件，它基本上是在单一编程语言（如梯形图程序）中的一系列的逻辑指令。每个程序必须有例行主程序，当触发任务调用相关程序时，它最先被扫描。而且，可指定一个故障程序和程序控制器的内存允许的多个附加子程序。可以创建多个标准程序库，而且在多机器或应用中可重复使用。

ControlLogix处理器模块采用标签来确认控制器中数据的存储位置。这种功能允许在控制程序中根据数据的功能给它们命名。如果想要组合数据，可以创建数组，数组是相同类型的标签组合。Logix-5550控制器使用如下标签类型：

- 基础标签——定义存储数据元素的存储区。
- 别名标签——可以引用已经由其他标签定义的存储区。
- 接收标签——引用来自于其他控制器的数据。

可以在逻辑输入前创建标签，在输入逻辑的时候输入标签名，或者用问号［?］代替标签名，过后再指定标签。这些选择使得在程序设计时有很高的灵活性。我们能创建符号标签名来访问独立的数据块或指向另一个标签。所有被使用的标签名与控制程序一起被下载到控制器的存储器中。接下来这些标签名在运行时用来对控制器数据提供访问。因此，不再需要关注控制器中数据的物理存储位置。当控制器被上载时，所有的标签名被收回，并提供部分文件程序。

当定义标签时，必须说明如下信息。

- 标签名，必须由字母或下划线（_）开始。标签名只能包含字母、数字或下划线，长度可达40字符，不能使用一连串的下划线和用下划线作后缀。
- 可选择的标签说明，其长度可达120个字符。标签说明没有被下载到控制器而是被储存在离线的项目文件中。
- 标签类型，基本、别名或接收。
- 数据类型，通过已预定的列表或用户定义数据类型获得。
- 创建标签作用域，可以选择控制器作用域或现存程序作用域中的一个。
- 显示风格，当采用编程软件监视标签时软件将显示可用类型选择。
- 是否想使标签对其他控制器来说是可用的，以及可以接收此标签的控制器的数量。

PLC-5和SLC-500存储全局数据表中的所有数据，它是以16位操作为基础的。通过指定数据的地址访问想要的数据。Logix控制器是建立在32位操作基础上的，支持程序中数据和在控制器中的所有任务的全局数据。Logix控制器也能和其他控制器分享数据，你可用标签而非地址访问你想要的数据。有两种预定义的数据类型：基本数据类型和结构体数据类型。基本数据类型（也称自动数据类型）包括：

基本数据类型	存储数据类型
BOOL	1位布尔类型
SINT	1字节整数
INT	2字节整数
DINT	4字节整数
REAL	4字节浮点数

使用基本类型数据为预定义功能创建结构体数据类型，其类型包括：

结构体数据类型	存储数据类型
CONTROL	数组指令的控制结构体
COUNTER	计数器指令的控制结构体
MOTION_INSTRUCTION	运动指令的控制结构体
PID	PID 指令的控制结构体
TIMER	定时器指令的控制结构体
AXIS	轴的控制结构体
MESSAGE	MSG 指令的控制结构体
MOTION_GROUP	运动组的控制结构体

基本标签所占的内存大小取决于数据类型。控制器存储数据的最小内存为 4B 或 32 bit。

- 一个 BOOL 基本标签由 1 位组成，且用第 0 位来储存信息（1 ~ 31 位不用）。
- 一个 SINT 基本标签由 8 位组成，且用 0 ~ 7 位储存它的数据（8 ~ 31 位不用）。
- 一个 INT 基本标签由 16 位组成，且用 0 ~ 15 位储存它的数据（16 ~ 31 位不用）。
- DINT 和 REAL 基本标签由 32 位组成，因此 0 ~ 31 位都被使用。

一个结构体是一组不同数据类型的组合，作为单一的整体，实现特定的目的。在 ControlLogix 控制器中有三种不同的结构体类型：预定义、用户定义和模块定义。结构存储一组数据，且结构中的每个成员可能是不同的数据类型。结构体中的成员通过标签名、点和成员名来寻址（标签名．元素名）。用户定义结构体补充了预定义结构体（例如定时器、计数器等）。一个用户定义结构可以包含任何基本数据类型（如布尔型、单字节整数、整数等）或结构体（预定义结构体或用户定义结构体）。

101

数组就是一系列的元素，每个元素具有相同的数据类型。数组允许把控制器存储区中数据类型相同的数据组合到一起。许多控制程序需要储存在表中的信息块在运行时能被移动。ControlLogix 控制器能满足这种需求，它可以创建多达 3 维的数组（即行、列和深）。数组中的单个标签是一个元素，元素可能是基本数据类型或一个结构体。元素从 0 开始一直到元素数量减 1。下表所示为一维数组的存储器布局，数组名为 Model_Data，数据类型为 DINT [8]，这个数组包含了编号从 0 ~ 7 的 8 个元素。

数组：　Model_Data

数据类型：DINT [8]

	Model_Data [0]
	Model_Data [1]
	Model_Data [2]
	Model_Data [3]
	Model_Data [4]
	Model_Data [5]
	Model_Data [6]
	Model_Data [7]

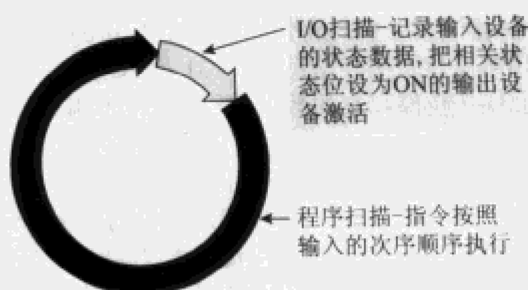
通常，在控制器中的数据被扫描，使所有的例行程序可访问读取和修改存储信息。因为这种系统能阻止例行程序库的创建当两个程序在同一个应用中时会有冲突。Logix 控制器通过提供程序和控制器数据包的方法解决了这个问题。创建的控制器全局数据可以被控制器内的所有逻辑访问，与 PLC-5 和 SLC 处理器内的数据表访问方式相同。然而，局部程序数据仅被单个程序中的例行程序访问。相同的标签名可以在不同的程序中作为局部变量，因为可以选择标签创建的作用域。

在 ControlLogix 控制器中，可以通过点对点连接实现控制器之间数据共享。当设计应

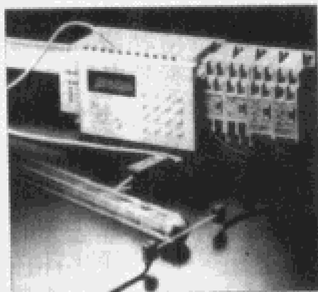
用时，可以通过背板为系统中其他控制器产生数据，也可以接收来自其他控制器的标签。这个特点允许你选择任一控制器接收和发送数据。同样地，多控制器能连接到任一产生的数据，这样就避免了发送包含相同数据的多个消息。

5.2 程序扫描

在每个操作周期中，处理器读取所有的输入信号，处理这些输入值，并根据用户程序激励或不激励相应的输出，这个过程称为*扫描*。图5-7说明了一个单一的PLC扫描过程，它由*I/O扫描*和*程序扫描*组成。由于输入可能随时变化，因此PLC要不断进行扫描处理。



(a) PLC的一次扫描



(b) 高速传感和机器控制应用，欧姆龙的S3D8传感控制器模块能接受3kHz的输入信号，反应速度为1ms（得到欧姆龙公司许可）

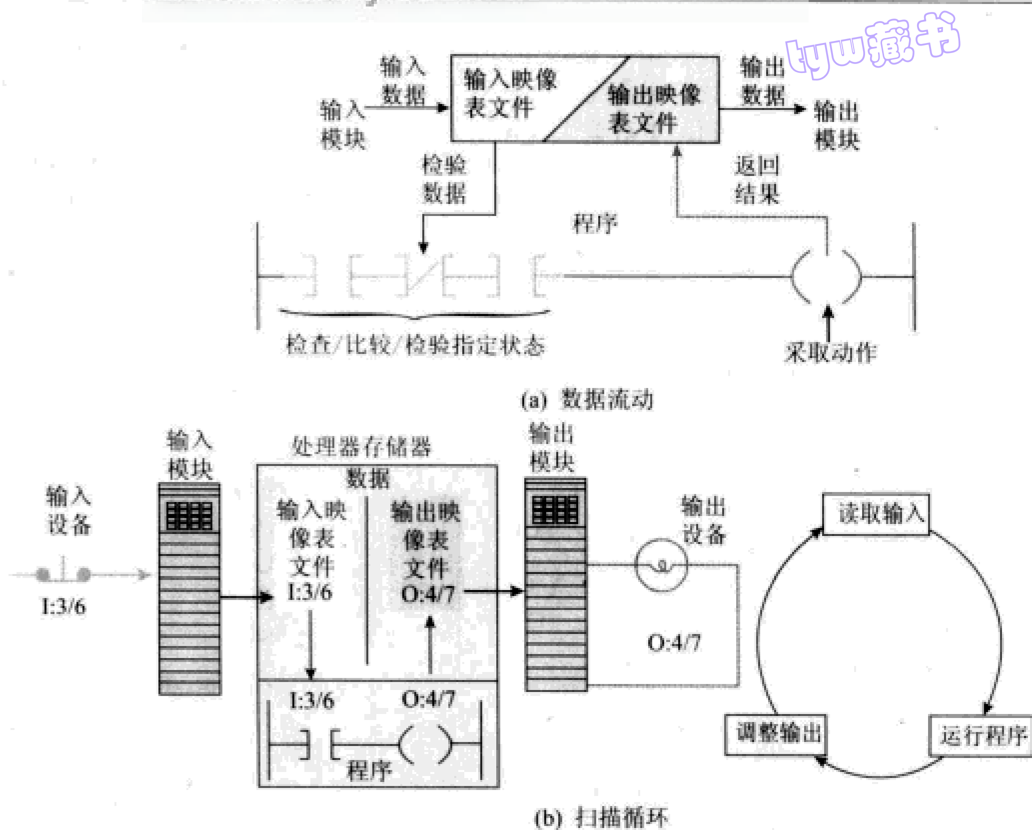
图 5-7

PLC的扫描时间的说明，表明了控制器对输入变化的反应速度。扫描时间的变化与程序内容和长短有关。单次扫描所需时间从1ms到20ms不等。若在扫描周期内，输入信号有两次变化，那么PLC将很可能从未检测到这种变化。例如，CPU扫描一个程序需要8ms，而输入触点每隔4ms接通或断开，那么程序可能并未反映触点状态的变化。若输入信号的变化发生在输入映像表文件的更新期间，那么CPU将检测到这个变化，但CPU并不会对每个变化都做出反应。 [102]

扫描通常是连续地，按次序地读入输入状态，判断控制逻辑，适时修改输出。图5-8说明了这个过程。当连接到输入地址I:3/6的输入装置被是闭合的，输入模块电路产生一个电压并且把输入映像表中的位I:3/6置1。在程序扫描期间，处理器检测输入位I:3/6是“1”。在这种情况下，由于输入I:3/6是“1”，则对应的梯级为真，然后在下一个I/O扫描期间处理器把输出映像表位O:4/7置“1”。连接到该终端的输出设备（灯）被激励。只要在处理器在运行模式，就一直重复这个过程。若输入装置是断开的，输入映像表中相应位被置“0”，因而这时的梯级条件为假。处理器将置输出映像表位O:4/7为“0”，输出装置被关断。 [103]

程序中的每条指令需要一定的时间执行，时间长短由指令决定。例如，处理器读取输入触点状态所需时间要比读计数器累加值的时间少。

不同的PLC制造商用两种基本的扫描方式来完成扫描功能（如图5-9所示），Allen-Bradley PLC通过梯级方式进行水平扫描。在这种系统中，处理器从程序中左上方的第一个命令开始，逐个梯级的水平扫描，检查输入和输出。AEG Modicon PLC通过列方式进行垂直扫描，在此系统中，处理器从梯形图左上端开始，按照逐列逐页的方式，进行垂直扫描，检查输入、输出指令。程序页被顺序执行。如果对PLC的扫描方式理解不正确，可能会引起编程错误。 [104]



(a) 数据流动

(b) 扫描循环

图 5-8 扫描过程

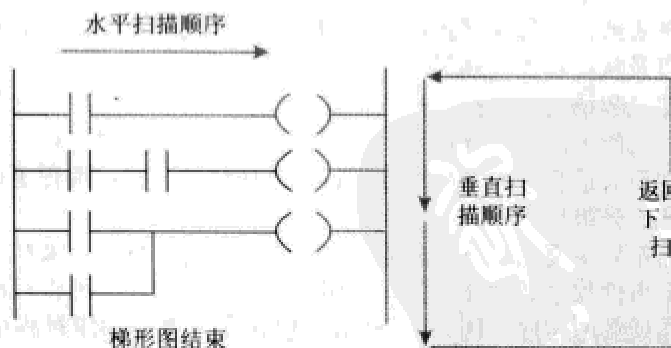


图 5-9 垂直或水平扫描方式

扫描用户程序所用的时间，还取决于微处理器系统的时钟频率。时钟频率越高，扫描速度就越快。

5.3 PLC 编程语言

PLC 的编程语言指的是用户与 PLC 进行信息交换的方法。最常用的三种语言结构是：梯形图语言、布尔语言和功能图。虽然对于不同类型的 PLC，它们每一种编程语言的结构相似，但不同的制造商提供的应用方式还是有所区别。只是这些差别较少，并且容易理解。

梯形图语言是迄今为止最常用的PLC编程语言。图5-10是梯形图程序和布尔程序的比较，图5-10a画出了类似于硬接线的原始继电器电路图，图5-10b是与之对应的控制器中的梯形图程序。注意到图中对输入和输出设备的编址采用自然属性方式，对不同类型的PLC可能不同。图5-10c为一系列典型的通用布尔状态，利用此状态也可以对原始的继电器电路进行编程。这种状态涉及基本的与、或和非逻辑门功能，以及典型的布尔方程式。

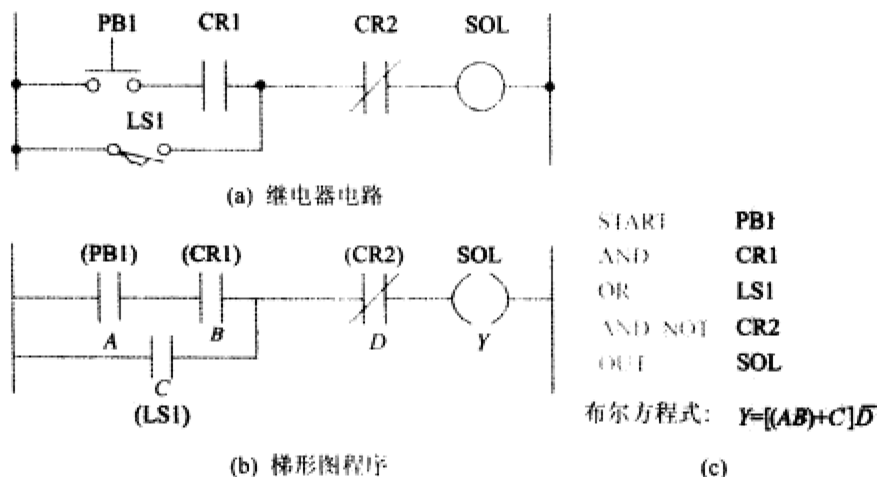


图5-10 PLC梯形图和布尔语言

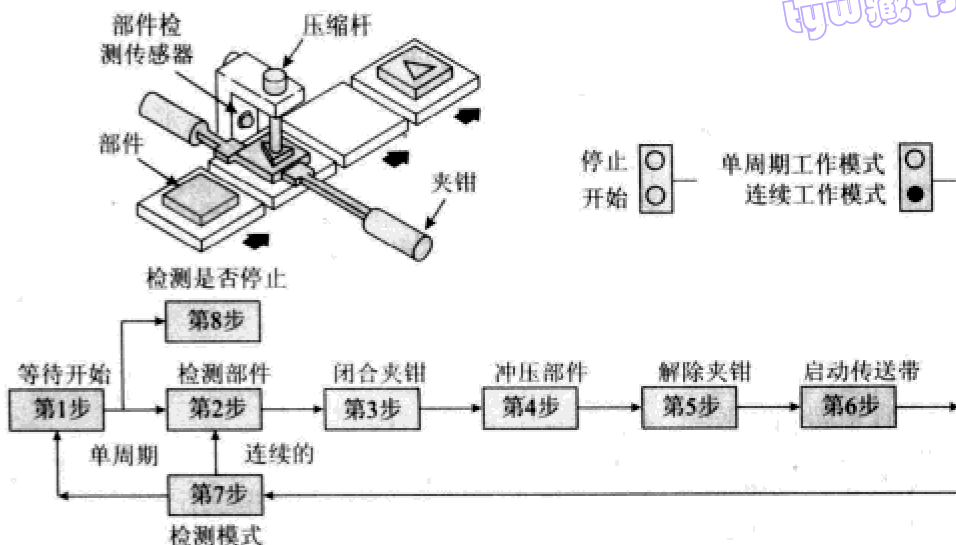
功能图编程系统最初在欧洲发展，被称为GRAFCET，它是一种用更结构化的方法控制系统编程。功能图编程语言用功能块（步和转换单元），通常通过布尔表达式进行控制。功能图程序是一种图表或是一种顺序控制过程的指定类型的流程图。它显示了这个控制过程可能采取的路径以及从一功能块到另一功能块所需的条件。

[105] 原则上，功能图编程语言允许把过程的描述变成实际的控制程序。这种方法有助于理解系统和问题定位。功能图的主要优点是：将逻辑块作为一个模块进行编程，且转移逻辑确保在任一给定时间只有适合的软件模块在运行。其他优点还有：编程更简化，扫描速度更快，可维护性更强，升级更容易。

任何过程或机器不管多复杂，均可用顺序或/和并行事件的组合来描述。整个程序可能相当复杂，但单个事件程序通常非常简单。用功能图编程指令，可将程序分为几个步或阶段，而非一个长的梯形图程序。图5-11显示了包含在简单的冲压过程中的任务和步骤的顺序。每一步表示了直接与机器的顺序操作相对应的事件。

当用功能图系统编程时，只需对步或阶段独立的编程，而不必考虑它们将如何影响程序的其他部分。支持这种编程类型的处理器知道程序的哪部分是激活的，并且仅扫描激活步，其余所有的程序不被扫描。这样，减少了程序扫描时间。[106]

在写梯形图程序时，为了确保过程按照正确的次序工作，需要花大量的时间和精力在互锁编程上。梯形逻辑的一个重要部分，专门用来解决互锁问题，或确保某些事情不会发生。功能图编程可以帮助避免这类问题的发生。处理器甚至不扫描未被激活的那部分程序，这样减少了复杂互锁请求的数量。除此之外，还使控制器更容易寻找故障。当故障发生时，能迅速判断出过程的哪一步或哪一阶段有问题，并且确切地知道机器的哪部分需要检查。



1. 操作员按下开始开关启动机器或者按下停止开关停止机器
2. 机器检测是否有部件，如果出现一个部件，处理器继续，否则，直到出现部件时传送带才开始移动
3. 部件用夹钳固定
4. 冲压部件
5. 解除夹钳，完成的部件被移走
6. 如果机器工作在单周期模式，过程停止；如果工作在连续模式，过程继续

图 5-11 包含在冲压过程中的步骤

Allen-Bradley PLC-5 处理器可以用多种功能图编程——顺序功能图（SFC）编程。PLC-5 SFC 使用户采用功能图元件编程。然而，继电器梯形逻辑（而不是布尔表达式）被用于逻辑控制。这允许 PLC 梯形图程序被分为多个部分或模块，便于故障查找和维护（如图 5-12 所示）。

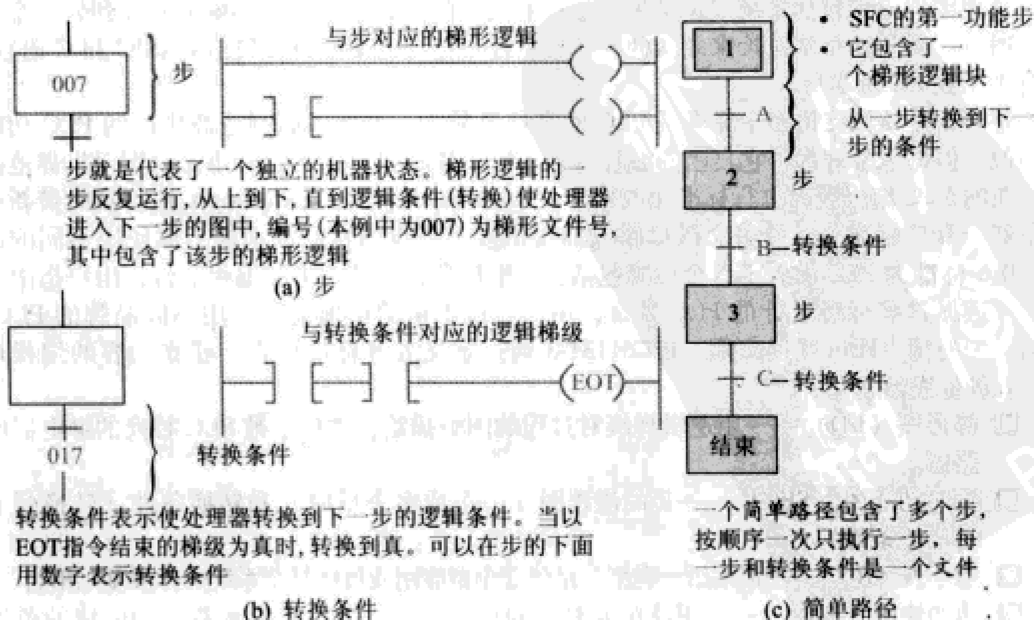
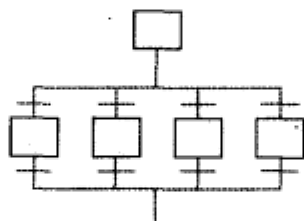
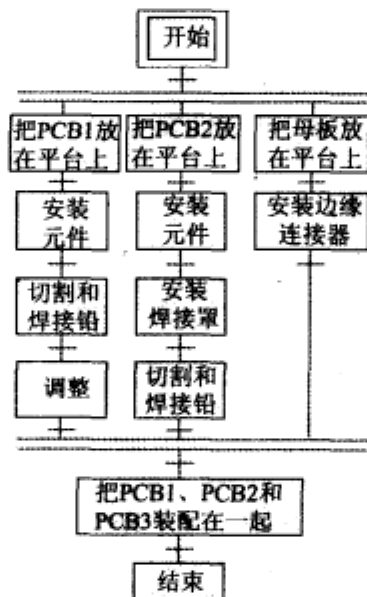


图 5-12 顺序功能图（SFC）编程



- 选择分支，包含了多个路径，处理器可以选择其中的一个
- 与或(OR)结构类似
- 画一个选择分支作为与单一水平线相连的平行路径
- 多个选择路径其中只有一个被选
- 每一路径开始的转换条件从左向右扫描，第一个为真的转换条件决定了路径的选择
- 例如依赖于创建编码，有一步必须是钻孔或磨光

(d) 选择分支



- 并行分支同时执行所有的并行步
- 同与(AND)结构类似
- 用连接到双水平线的平行路径表示并行分支
- 在结构中所有的路径都激活
- 两个或多个路径至少扫描一次
- 处理器完成并行分支的一步，然后转换到下一步
- 通常路径的转换条件在分支之外
- 当完成了在每一分支的每一步至少扫描一次时，并且转换条件为真，处理器完成了并行分支的运行
- 例如，假设要在电路板PCB1上安装元件，并且在PCB2上焊接元件，母板已经完成，只等待其他两个板子扫描的逻辑只是在PCB1上安装元件和在PCB2上进行焊接

(e) 并行分支

图 5-12 （续）

其他编程语言包括系统被定义为一系列的状态的状态图，从一个逻辑元件到另一个元件是通过一系列的布尔决策来激活的。还有其他像传统的编程语言包括 BASIC 和 C 语言，来自于工业 PC 机本身。

整个工业领域支持多个品牌的 PLC 确实是个梦魇。虽然不同制造商生产的 PLC 功能很相似，但每个品牌都有它自己的编程规则。PLC IEC61131-3 国际标准是试图对世界范围内存在的 PLC 编程语言进行标准化的代表。Allen-Bradley 的 ControlLogix 控制器的最新产品是基于 IEC 标准的。大多数 PLC 供应商提供或多或少的相同的功能，但使用不同的语法。IEC 标准为 PLC 设置了 5 个标准包括：一般信息，硬件要求，编程语言，用户指南和通信。遵循这些标准设计的 PLC 系统，用户经过少量的培训就能使用不同品牌的 PLC，并用很少的精力即可互换应用。IEC61131-3 通过定义五种标准语言，建立相容的编程环境。五种标准语言是：

107
108

- ❑ 梯形图（LD）——用逻辑梯级对过程的图解描绘，类似于被 PLC 替换的继电器电路图。
- ❑ 顺序功能图（SFC）——步的流程图（一个或多个反应）和转换条件（转换到下一步的条件）。
- ❑ 指令表（IL）——汇编语言类型，是建立小型应用或最优复杂系统的标准基础语言。
- ❑ 功能块图（FBD）——用简单或复杂的功能块的图表处理器流程，功能块的范围从模拟 I/O 到闭环控制、算法和诊断。

❑ 结构文本（ST）——IEC 标准的高级语法编程语言，使用 if 和 then 语句。

5.4 继电器类型指令

梯形图语言是用来创建控制器程序的一系列基本指令。这些梯形指令符号被排列实现期望的输入到 PLC 的存储器中的控制逻辑。由于指令系统由触点符号组成，梯形图语言也被称为*触点符号*。

触点和线圈是梯形图指令系列中的基本表达式。用来转换继电控制逻辑到触点符号逻辑的三种基本符号是：检查是否闭合（EXAMINE IF CLOSED）、检查是否断开（EXAMINE IF OPEN）和输出激励指令（OUTPUT ENERGIZE）（参看图 5-13 ~ 图 5-16 中给出的符号和例子）。

符号

类似于继电器常开触点，用这些指令，使处理器检查触点是否闭合

表示控制逻辑的典型输入

输入可以使连接的开关或按钮，可以是连接输出的触点，或者是内部输出的触点

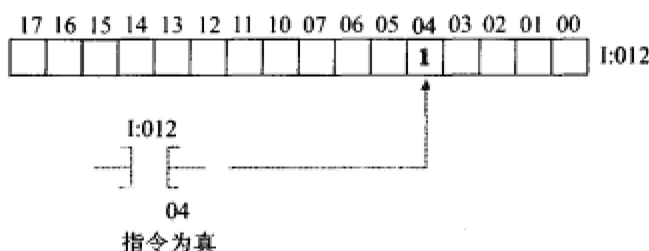
具有位地址

位的状态或者为1(ON)或者为0(OFF)

检查状态位的ON状态

如果状态为1，指令为真

如果状态为0，指令为假



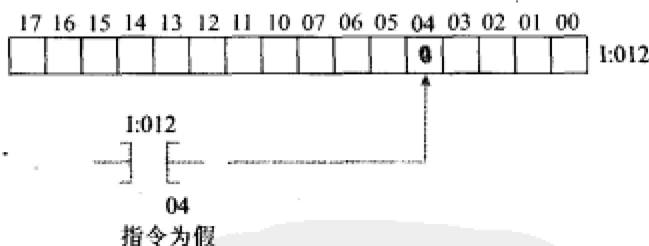


图 5-13 检查是否闭合（XIC）指令

符号

类似于继电器常闭触点，用这些指令，使处理器检查触点是否断开

表示控制逻辑的典型输入

输入可以使连接的开关或按钮，可以是连接输出的触点，或者是内部输出的触点

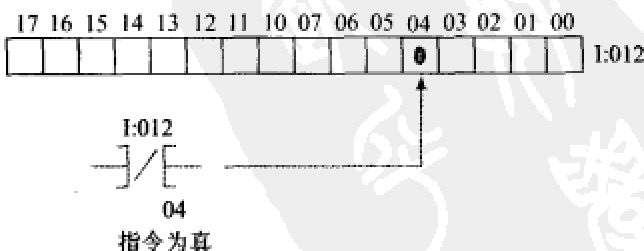
具有位地址

位的状态或者为1(ON)或者为0(OFF)

检查状态位的OFF状态

如果状态为0，指令为真

如果状态为1，指令为假



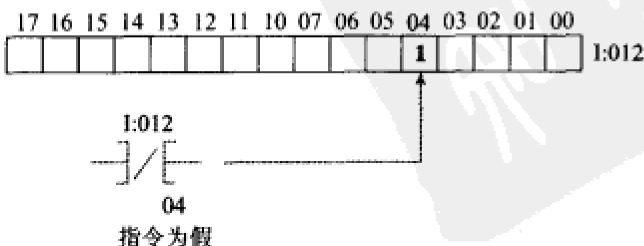
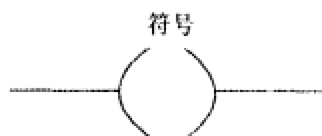


图 5-14 检查是否断开（XIO）指令

符号



类似于继电器线圈，当梯级中的XIC和XTO指令为真时，处理器使指令变为真（类似于把线圈接通）

表示由组合的输入逻辑所控制的输出

可以连接输出设备或内部输出（内部的继电器）

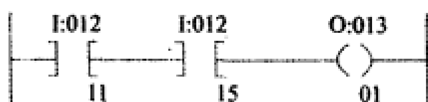
如果任何从左到右的输入状态为真输出被激励

当梯级为真时，输出激励指令的取址状态位被置1

当梯级为假时，输出激励指令的取址状态位被置0

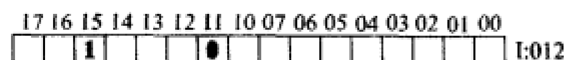


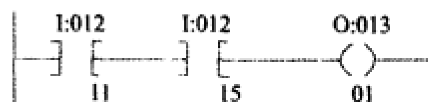




输出激励指令为真

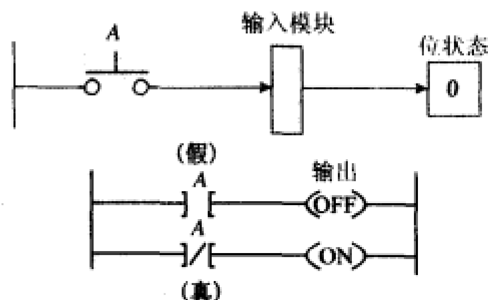




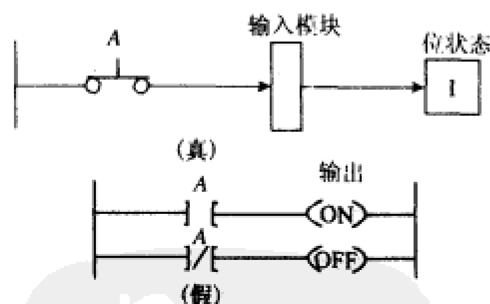


输出激励指令为假

图 5-15 输出激励（OTE）指令



按钮没有按下



按钮被按下

图 5-16 状态位举例

[109-110] 梯形图程序的主要功能是控制基于输入状态的输出，这个控制是通过采用梯形梯级来实现的。通常，梯级包含了用触点指令表示的一系列输入状态和在梯级最后用线圈符号表示的输出指令（如图 5-17 所示）。每个触点或线圈符号都涉及一个确认计算对象和控制对象的地址号，只要那个状态需要评估，相同的触点指令可以在整个程序中使用。为了激活或激励一个输出，至少有一个从左到右的触点路径必须是闭合的。一个完整的闭合路径被称为具有逻辑连续。当逻辑连续存在于至少一条路径时，梯级条件为真（TRUE），否则梯级条件为假（FALSE）。

在控制器操作期间，处理器决定数据文件中位的 ON/OFF 状态，判断梯级逻辑，并且根据梯级的逻辑连续性改变输出状态。

[111] 更特殊地，输入指令建立条件，在此条件下，处理器决定输出指令为真或为假，如下所述：

- 当处理器发现在一个梯级中有输入指令的连续通路时，OUTPUT ENERGIZE（OTE）输出指令将变为（或保持）真，这时我们称梯级条件为真（TRUE）。

- 当处理器没有发现在梯级中有输入指令的连续通路，OTE 输出指令将变为（或保持）假，这时我们说梯级条件为假（FALSE）。

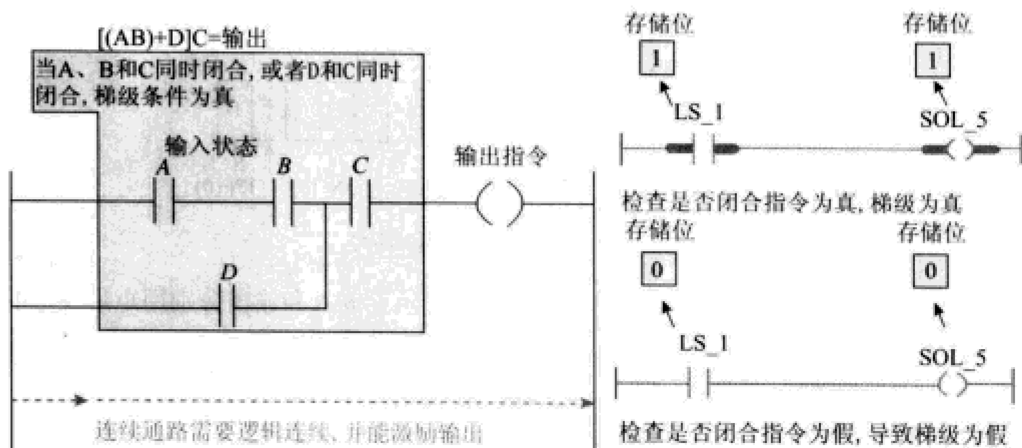


图 5-17 逻辑梯级

5.5 指令寻址

为了完成输入继电类型指令，你必须分配一个地址号给它，这个地址号将指明输入端和输入设备以及输出端子和驱动的外部设备的连接状况。

真正的外部输入和输出的寻址和内部状态一样，取决于使用的 PLC 模块。这些地址可以用十进制、八进制或十六进制表示，具体由使用的 PLC 的数字系统决定。图 5-18 所示为 Allen-Bradley SLC-500 控制器的典型寻址形式。参考 PLC 的编程手册才能确定具体的形式，这是因为就像制造商与制造商不同一样，模块与模块之间的形式也不同。

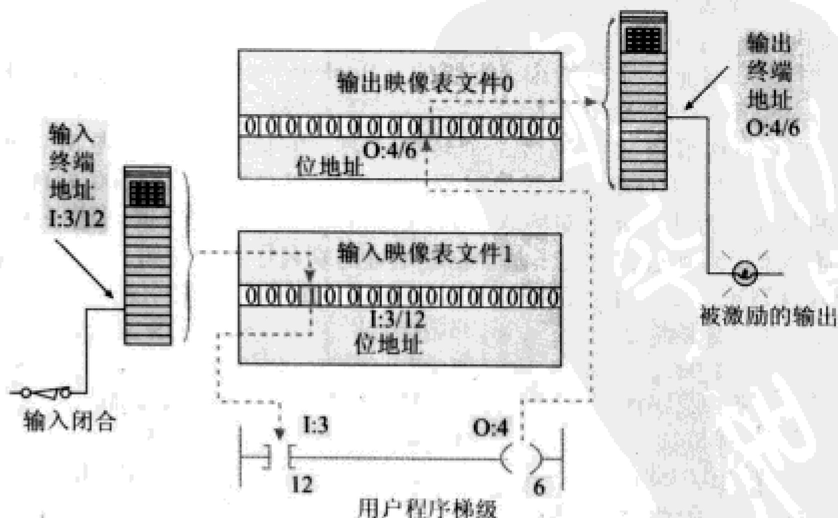


图 5-18 地址识别处理器数据文件中的一个存储位置，数据文件存储位的 ON/OFF 状态

地址识别一个指令的功能，并且将它连接到部分存储器数据文件的特定位上。图 5-19 是 16 位字结构和它的位值。

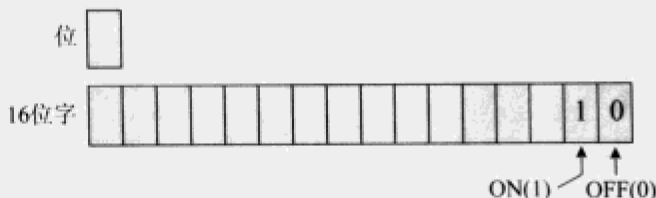


图 5-19 一个 16 位字的结构

I/O 地址分配包含在 I/O 连接图中，如图 5-20 所示。输入和输出分别用矩形菱形的图表表示。

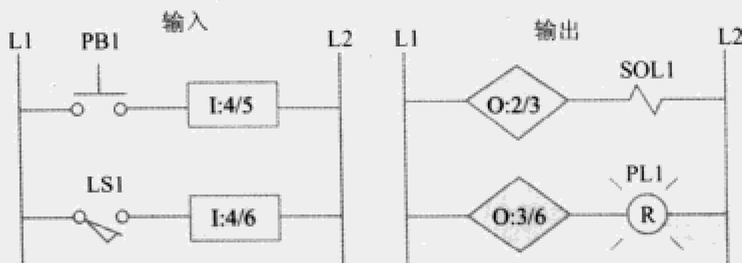


图 5-20 I/O 连接图

112 用 Allen-Bradley ControlLogix 控制器，可使用符号标签名（可达 40 个字符长度）来分配地址，它遵从 IEC61131-3 标准。只要对 I/O 组态并赋给标签名一次，软件中所有的应用都会变成相同的标签名。

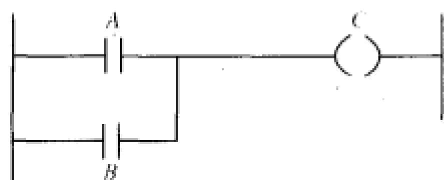
借助 I/O 组态向导，通过简单地单击你想要创建的模块类型，就能实现对 I/O 模块的组态。模块创建向导通过一系列的对话框引领你完成 I/O 模块的创建和组态。通过通信节点名、物理槽号、I（表示输入）、O（表示输出）和 I/O 模块接收与发送数据的参考数据的组合来创建标签。寻址一个位指令，需要位（Boolean）地址。在基本标签中，位是对二进制的标签名引用，例如 Switch-1。

RSLogix SLC-500 编程软件包括一个转换工具，它能把 PLC-5 或 SLC-500 文件用 Logix 控制器转换使用。I/O 数据表转换过程力求保持 PLC-5 和 SLC-500 处理器中输入和输出映像表的布局。为此，分别对输入（I）数据和输出（O）数据创建一个一维数组。为了保留原始地址，转换过程创建了基于物理地址的别名标签。在如下所示的例子中，记住 PLC-5 用八进制对 I/O 寻址，而 SLC-500 ControlLogix 控制器用十进制。

控制器	原始地址	转换地址	别名标签名
PLC-5	I:007	I [7]	I_07
	I:021/05	I [17]. 05	I_021_Bit05
	O:010	O [8]	O_010
	O:035/15	O [29]. 13	O_035_Bit015
SLC-500	I:007	I [7]	I_07
	I:021/05	I [21]. 05	I_21_Bit05
	O:010	O [10]	O_010
	O:035/15	O [35]. 15	O_35_Bit015

5.6 分支指令

分支指令被用来创建输入条件指令的并行路径。这允许在一个梯级中有两个以上的输入条件（或逻辑）的结合来建立逻辑连续。图 5-21 为一个简单的分支条件。只要指令 *A* 或 *B* 中的一个为真，则梯级为真。 113



增加一个分支

通过单击指令工具栏中的这个指令图标在梯形图中创建一个分支。如果光标在一个指令上面，分支立即在这个指令的右边创建。如果光标放在梯级标号上，分支被创建于梯级上

图 5-21 并行路径（分支）指令

根据并行分支信息的输入分支用于应用程序，可以允许有更多的输入条件组合，如图 5-22 所示。若在这些并行路径中，至少有一条并行分支形成逻辑为真的路径，则梯级逻辑连续。若并行路径无一条形成逻辑为真的路径，那么梯级逻辑不连续，且输出指令逻辑将不为真（TRUE）。这个例子中，*A* 和 *B*，或 *C* 都能提供一个逻辑为真的路径。

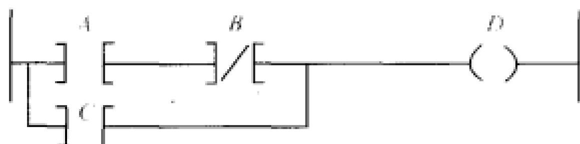


图 5-22 并行输入分支

对于大多数 PLC 模块，梯级的输入和输出部分都可以创建分支。在输出分支中，可以用一个逻辑为真的路径控制多个输出，如图 5-23 所示。当存在一条逻辑为真的路径时，所有并行输出变为真（TRUE）。这个例子表明，*A* 或 *B* 都可以提供一个逻辑为真的路径给驱动 *C*、*D* 和 *E* 三个输出指令。

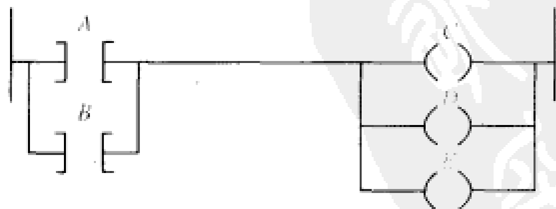


图 5-23 并行输出分支

附加的输入逻辑指令（条件）可用于输出分支编程，以加强输出的条件控制。如图 5-24所示，当存在的路径包含了输出分支中额外的输入条件的逻辑为真时，分支变为真（TRUE）。在本例中，*A* 和 *D* 或 *B* 和 *D* 都能给输出 *E* 提供一条逻辑为真的路径。

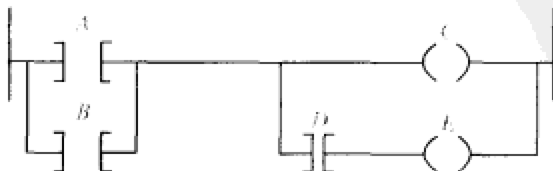


图 5-24 带有条件的并行输出分支

[114] 输入和输出分支能被嵌套以避免冗余指令，同时加快处理器扫描速度。图5-25所示为嵌套的输入输出分支，一个嵌套分支在另一个分支中开始或结束。

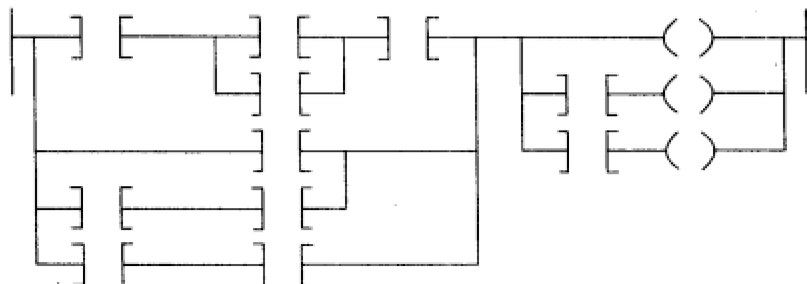


图5-25　嵌套的输入和输出分支

某些PLC模块，不能直接在一个分支电路中编程分支电路或嵌套分支。然而，可以对逻辑与之等同的分支条件进行编程。图5-26是包含了嵌套触点 *D* 的电路的一个例子。为了获得需要的逻辑，可按图5-27所示的电路进行编程，触点 *C* 的两次使用，避免了触点 *D* 的嵌套。通过指令的重复使用制造并行共同物，嵌套分支能被转换成无嵌套的分支。

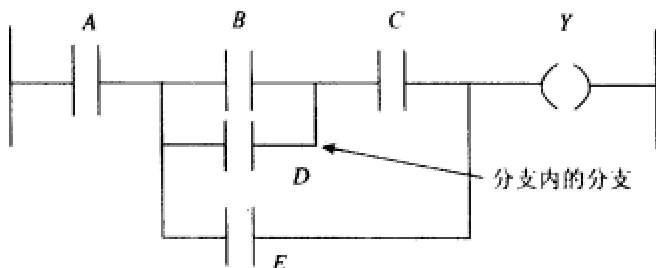


图5-26　嵌套触点程序

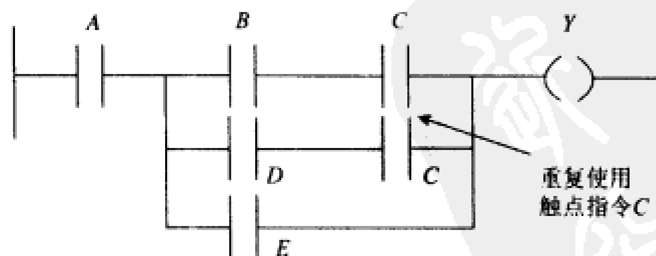


图5-27　避免了触点嵌套的程序

实际上，某些PLC制造商对允许的串联单元，并行分支或输出没有限制。而另一些制造商，在梯形图程序中一个梯级串联触点的数量，以及并行分支数量可能有限制。同[115]样，对某些PLC还有另外的限制：每个梯级仅有一个输出，且输出必须连接到梯级的末端。梯级数量上唯一的限制是存储器大小。图5-28显示了一个典型PLC矩阵限制图。最大有7个并行线路，每个梯级允许有10个触点串联。

对分支电路编程的另一个限制是PLC不允许用垂直触点编程。此限制的典型例子如图5-29用户程序中的触点 *C*。为了获得要求的逻辑，电路将重新编程如图5-30所示。

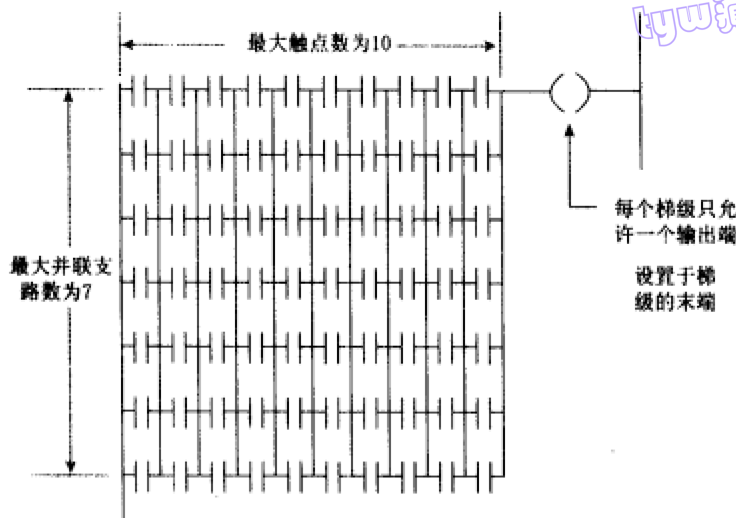


图 5-28 典型 PLC 矩阵限制图。准备的限制性取决于所使用的 PLC 的特殊类型。当编写的程序中串联单元、并联支路或输出端的数量多于允许的数量时，将导致显示一个错误的信息

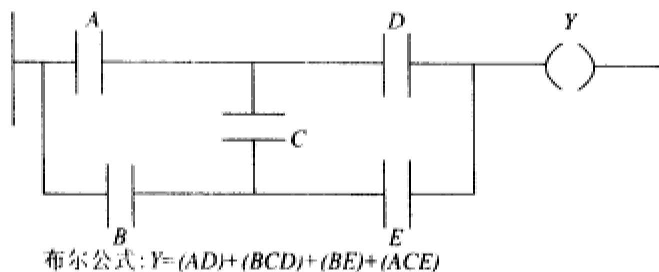


图 5-29 具有垂直触点的程序

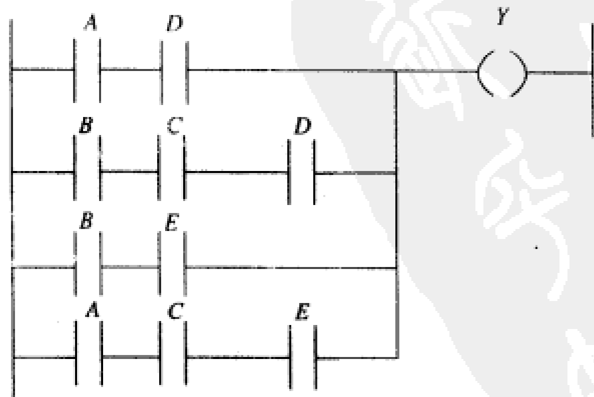


图 5-30 重新编程以避免垂直触点

处理器仅仅从左到右检查梯形图是否有逻辑连续存在，而决不允许从右到左。这样用户程序就会出现类似图 5-31 所示的问题，若按如图所示编程，触点组合 FDBC 将被忽略。为了获得要求的逻辑，电路将重新编程如图 5-32 所示。

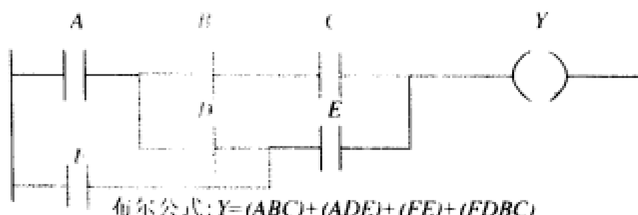


图5-31　原始电路

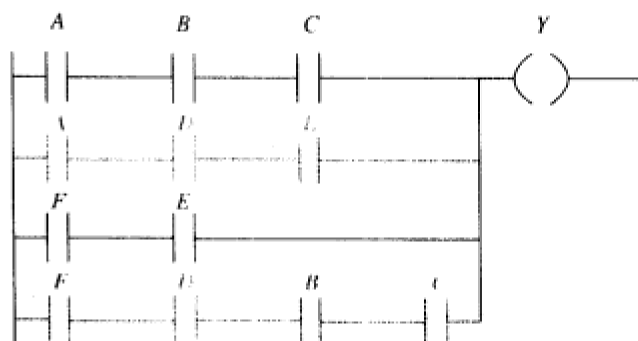


图5-32　重新编写的电路

5.7　内部继电指令

大多数PLC都分配了内部存储位区域，这些存储位也称为内部输出、内部线圈、内部控制继电器或简单内部位。内部输出操作起来正像由程序逻辑控制的任何输出；然而，这些输出是按内部目的而严格使用的。换言之，内部输出不能直接控制输出装置。

用内部输出的优点是，在程序中很多情况下输出指令被在程序中使用，但它和现场装置没有物理连接。若没有物理输出连接到位地址，则此位地址可被用作一个内部储存点。内部储存位或点可通过用户编程，完成继电功能，而不占物理输出。这样内部输出就最小化了所需的实际的输出卡。

[116]

当一个电路要求比梯级允许的串联触点多时，就要用到内部控制继电器。图5-33所示的电路仅允许有7个系列触点，而预程序逻辑要求12个触点。为了解决这个问题，触点被分解成两个梯级。第一个梯级包括7个要求的触点用内部继电器进行编程，内部继电器地址为第2个梯级中第一次检查是否闭合（EXAMINE IF CLOSED）的地址，其余5个触点下面是数字输出。内部储存位的优点是不浪费物理输出空间。

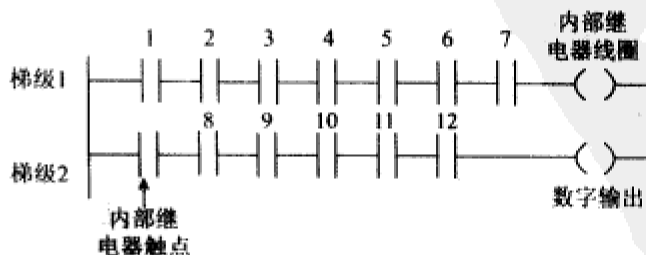


图5-33　内部控制继电器

5.8 检查是否闭合与检查是否断开指令

用检查是否闭合指令（EXAMINE IF CLOSED（XIC））编写的简单程序如图 5-34 所示，此图显示了一个硬接线电路和提供相同结果的用户程序，注意到常开（NO）和常闭（NC）按钮都用 EXAMINE IF CLOSED（XIC）符号代表，这是因为输入（常开或常闭）的正常状态对控制器不重要。关键是是否需要触点闭合来激励输出，于是就要用 EXAMINE IF CLOSED指令。由于在图 5-34 中的 PB1 和 PB2 必须闭合才能激励指示灯，因此对两者都需使用 XIC 指令。

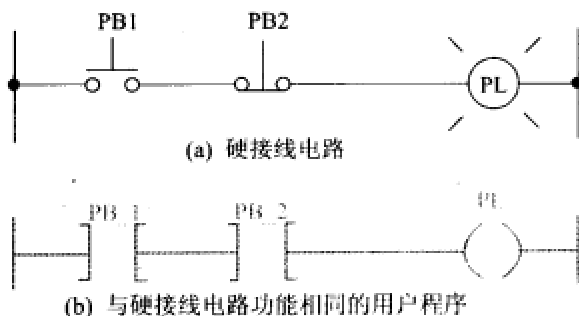


图 5-34 用 XIC 指令编程的简单程序

用检查是否断开指令（EXAMINE IF OPEN（XIO））指令编写的简单程序如图 5-35 所示，再次显示了硬接线电路和用户程序。在硬接线电路中，当按钮是断开时，继电器线圈 CR 未被激励，触点 CR1 闭合指示灯接通。当按钮闭合时，继电器线圈 CR 被激活，触点 CR1 断开指示灯熄灭（OFF 态）。在用户程序中，按钮用 EXAMINE IF OPEN 指令表示。这是因为当外部按钮是断开时，梯级必须为真（TRUE），当外部按钮闭合时，则梯级必须为假（FALSE）。

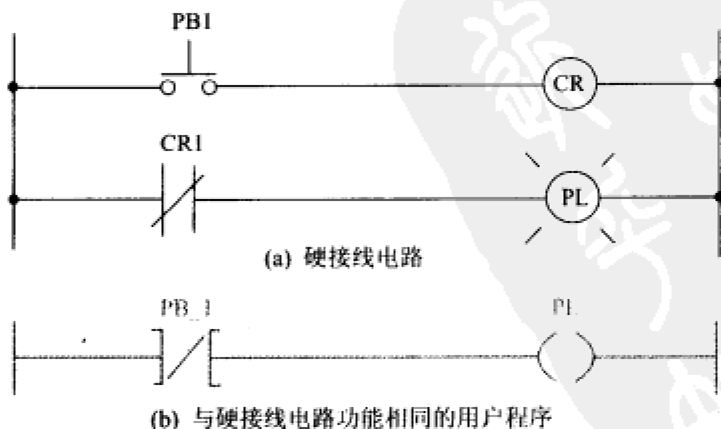


图 5-35 用 EXAMINE IF OPEN(XIO)指令编写的简单的程序

用 EXAMINE IF OPEN 指令来表示按钮以满足功能要求。不必考虑按钮的常开（NO）或常闭（NC）按钮的机械动作。一定要记住用户程序不是实际的电子电路，而是逻辑电路。事实上，当确定一个输出时，我们感兴趣的是梯级的逻辑连续性。

图 5-36 是用 XIC 和 XIO 指令编写的简单程序。逻辑状态（0 或 1）指示指令为真

（TRUE）或假（FALSE），是控制器操作的基础。图中表明，输出的 ON/OFF 状态由梯级中输入状态的改变决定。

数据表位的状态	指令的状态		
	XIC 检查是否闭合 —] [—	XIO 检查是否断开 —]/[—	OTE 输出激励 —( )—
逻辑0	假	真	假
逻辑1	真	假	真

时间	输入		输出	位状态		
	XIC	XIO	OTE	XIC	XIO	OTE
t_1（初始）	假	真	假	0	0	0
t_2	真	[illegible]	变为真	1	0	1
t_3	真	假	变为假	1	1	0
t_4	假	假	保持假	0	1	0



图 5-36　用 XIC 和 XIO 指令编程的简单程序

5.9　梯形图的输入

虽然基于 DOS 的编程软件现在仍然可用，但当今大多数编程软件包都是在 Windows 环境下进行操作的。例如，Allen-Bradley 的 RSLogix 软件包是 Windows 编程包，用来编辑梯形图程序。此软件有各种不同的版本，可用来对 PLC-5、SLC-500、ContrlLogix 和 Micrologic 系列处理器编程。新增加的特点是 RSLogix 与先前创建的基于 DOS 编写的程序兼容。可输入用 DOS 产品开发的项目或从 RSLogix 输出工程项目。

输入梯形图程序或进行实际的编程，通常是用电脑键盘或手持式编程设备来完成。因为硬件和编程技技术随制造商的不同而有所变化，因此参考编程手册以便决定特定 PLC 的指令输入是必要的。

输入程序的一种方法是通过手持键盘，键盘上通常有继电符号和特殊的功能键以及用于输入地址的数字键。也有一些字母数字混合键，用于其他特殊的编程功能。在手持式单元中，按键很小且具有多个复杂的功能。多功能键就好像计算器上的第二功能键。

个人计算机是现在最常用的编程设备，它通过相关的 PLC 软件与特定的 PLC 模块相匹配。

图 5-37 为 RSLogix 软件的主窗口，通过 Windows 环境，不同的屏幕，工具栏以及对话框被用来导航。理解各种屏、工具栏和窗口的用途以便最有效地使用该软件是很重要的。这些信息可通过软件参考手册获得，以适应特定的 PLC 类型，同时也使得用软件编程时会更得心应手。

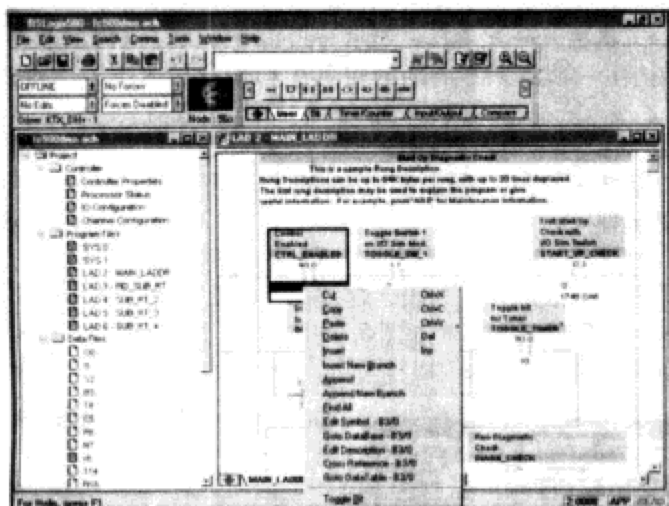


图 5-37　RSLogix SLC-500 的主窗口

图 5-38 表示选择了位指令的典型指令工具条。为了在梯级中放置一条指令，单击工具条上的相应图形符号且把它直接拖放在梯级图中的梯级上。落点显示在梯级上以帮助对指令定位。而且，指令还可从项目中的其他梯级中拖过来。可用几种不同的方法对指令进行编址：可以通过手动进入一个地址，或者从数据文件或其他指令中拖动地址。

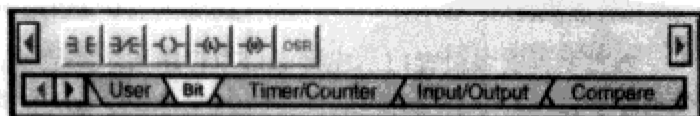


图 5-38　选择了位指令的典型指令设置工具条

当使用 RSLogix500 软件工作时，需要用到以下窗口。

- **主窗口**——每次你创建一个新项目或打开已有项目时，此窗口就会打开。与主窗口有关的一些特征如下：
  - **窗口标题栏**——标题栏放置在窗口最上端，显示程序名以及打开的文件名。
  - **菜单栏**——放置在标题栏下方，它包含与菜单有关的关键字，单击关键字可打开相应菜单。
  - **窗口工具栏**——当单击窗口工具栏按钮时，执行标准窗口命令。
  - **程序/处理器状态工具栏**——此工具栏包含 4 个下拉列表，以确定当前处理器操作模式，当前在线编辑状态，是否强制和启动。
  - **项目窗口**——显示项目树中的文件夹列表。
  - **项目树**——项目树是包含在当前项目中所有的文件夹以及相关文件的直观表示。从项目树中，你可打开文件、创建文件、修改文件参数、复制文件、隐藏或不隐藏文件、删除文件和修改文件名。

- **结果窗口**——此窗口显示搜索或检验操作的结果。检验操作用来检查梯形图程序的错误。
- **激活键**——用来鉴别当前哪个程序被激活。
- **状态栏**——此栏包含与当前文件有关的信息。
- **分割栏**——用来分裂梯形窗口以便显示两个不同的程序文件或梯级组。
- **标记指令工具栏**——此工具栏显示一组指令作为一组标记分类。
- **指令调色板**——此工具栏将所有可利用的指令显示在一个表格中，使指令选择更容易。
- **梯形窗口**——显示当前打开的梯形程序文件，用来开发和编辑梯形图程序。
- **梯形窗口性能**——此窗口允许你改变梯形图程序的显示，与之相关的地址和文档。

❑ **选择处理器类型**——编程软件需要知道要与用户程序连接的处理器。选择处理器类型窗口（如图 5-39 所示），包含了 RSLogix 软件可以对其编程的不同处理器的一个列表，你只需滚动下拉列表，直到找到你正使用的处理器并选中它。

❑ **I/O 组态**——I/O 组态窗口（如图 5-40 所示），让你从所有列表中单击或拖放一个模块，放到你组态的一个槽中。

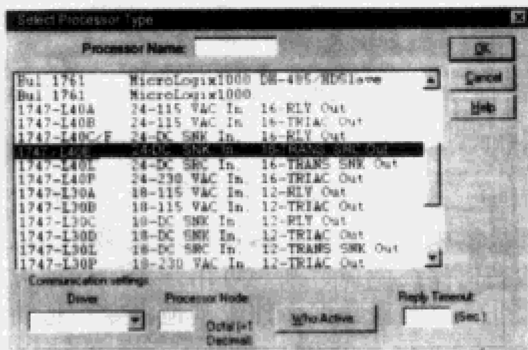


图 5-39 选择处理器窗口

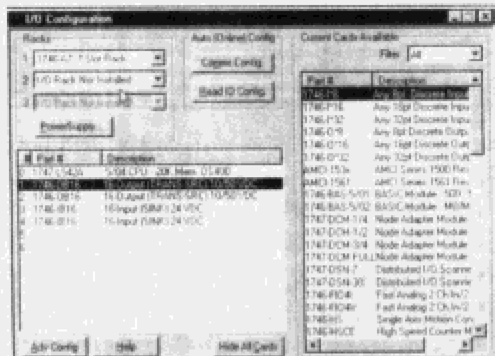


图 5-40 I/O 组态窗口

120 ~ 121

❑ **数据文件**——数据文件窗口包含用于连接梯形图程序指令的数据，包括输入输出文件以及定时器、计数器、整数、位文件。图 5-41 是位文件 B3 的一个例子，它用来作为内部继电器。注意，从这个文件开始的所有地址都从 B3 开头。

继电器梯形逻辑是一种图形化的编程语言，可贴切地表示一个硬连线继电系统。它提供了 PLC 控制可考虑的优越性。不仅仅是因为相当直观，特别适用于具有继电器经验的技术人员，而且对 PLC 工作在在线模式，正在实际实施控制时也特别有效。屏幕上各种指令梯级变亮，能够实时指示逻辑状态和逻辑连续性，使逻辑的操作变得明显，如图 5-42 所示。

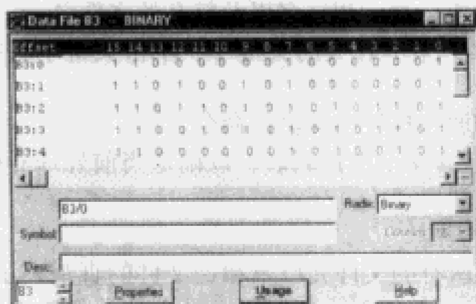


图 5-41 数据位文件 B3 窗口

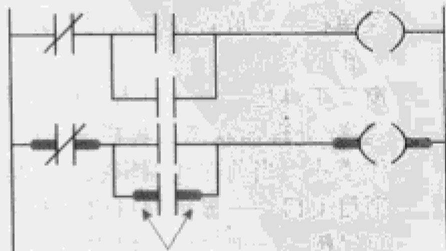


图 5-42 监视梯形图程序

对大多数 PLC 系统，每个 EXAMINE IF CLOSED 和 EXAMINE IF OPEN 触点、每个输出、每个分支 START/END 指令需要用户存储器的一个字。在编写程序时可参考 SLC-500 控制器性能指标确定使用的指令字的编号以及随着程序运行剩余的编号。

5.10 操作模式

处理器通常有两种基本的操作模式：*程序模式*和不同的*运行模式*。不同的操作模式和访问它们方法根据制造商的不同而有区别。一些常用的操作模式解释如下：

程序模式：程序模式被用来输入一个新的程序，编辑或修改已存在的程序，上传文件，下载文件，打印输出程序或在程序中改变任何软件组态文件。当 PLC 置于程序模式时，来自 PLC 的所有输出被强制关断，不管它们的梯级逻辑状态，且梯形 I/O 扫描顺序停止。 [122]

运行模式：运行模式用来执行用户程序。输入装置被监视，输出装置根据逻辑状态被激励。在所有指令已被写入一个新程序或对已有程序进行了修改后，处理器进入运行模式。

测试模式：测试模式用来操作或监视用户程序而没有激励任何输出。处理器仍然读取输入信号，执行梯形图程序和修改输出状态表文件，但没有激励输出电路。这种功能常用在开发或编辑程序之后，在允许 PLC 去操作实际输出之前，测试程序的执行情况。各种测试模式包括：*单步测试模式*，处理器执行选中的单个梯级或一组梯级；*单扫描测试模式*，处理器只进行一次扫描或循环操作；*连续扫描测试模式*，处理器连续检查程序，或进行故障诊断。

远程模式：有些处理器有三个位置开关来改变处理器操作模式。处于运行位置，所有的逻辑被处理且 I/O 被激励。处于程序位置，所有的逻辑处理被停止，且 I/O 未被激励。远程位置允许 PLC 远距离在程序模式和运行模式之间转换，通过连接到 PLC 处理器的个人计算机进行操作。进入远程模式之前，PLC 保持原来的操作模式。 [123]

思考题

1. 简述典型 PLC 存储器图中用户程序部分的作用。
2. 简述典型 PLC 存储器图中数据表部分的作用。
3. a. 输入映像表文件中储存什么信息？
   b. 信息以什么形式储存？
4. a. 输出映像表文件中储存什么信息？
   b. 信息以什么形式储存？
5. 列举出在单个 PLC 程序扫描中涉及的事件的次序。
6. 画出图 5-43 继电图对应的梯形图程序，写出布尔程序以及布尔方程。

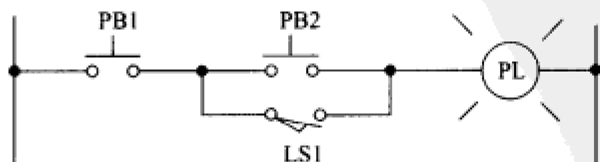


图 5-43

7. 画出常开（NO）触点、常闭（NC）触点、线圈的图形符号并说出与之对应的指令。
8. a. EXAMINE IF CLOSED 或 EXAMINE IF OPEN 指令表示什么？

b. OUTPUT ENERGIZE 指令表示什么？
c. 一个 EXAMINE IF CLOSED 指令的状态位是“0”，表示什么含义？
d. 一个 EXAMINE IF OPEN 指令的状态位是“1”，表示什么含义？
e. 在什么条件下，输出激励指令的状态位是“0”？

9. a. 描述梯形逻辑梯级的基本组成。
   b. 如何区分梯级中的触点和线圈？
   c. 梯级在何时被认为是真（TRUE）或逻辑连续？
10. 包含在某些5位PLC地址格式中的两种地址是什么？
11. 内部控制继电器的功能是什么？
12. 用常开限位开关编程控制一个电磁线圈，是用 EXAMINE IF CLOSED 还是 EXAMINE IF OPEN 触点指令？ 124
13. 简述如下PLC操作模式：
    a. PROGRAM b. TEST c. RUN
14. 数据文件是如何组织的？
15. 列出8种不同类型的数据文件。
16. 比较水平（行）扫描和垂直（列）扫描方式检测输入输出指令的方式。
17. 列写出由国际标准规定的可编程控制器的五种PLC标准编程语言，并对每一种做简单解释。
18. 解释 TRUE 梯级条件和 FALSE 梯级条件的含义。
19. 解释输入分支的功能。
20. ControlLogix 处理器的存储器组织结构与传统的 PLC-5 和 SLC-500 处理器有何不同？
21. ControlLogix 控制器中，一个项目的三个主要部分是什么？它们各自的功能如何？
22. 与 ControlLogix 控制器有关的两种任务类型是什么？
23. 在 ControlLogix 控制器中，对储存梯形逻辑位置的终端是什么？
24. ControlLogix 处理器用什么来确定存储器储存数据的位置？
25. 在 ControlLogix 控制器中，使用了哪三种类型的标签？每种标签的内容是什么？
26. 什么数据类型储存在如下基本数据类型中？
    a. BOOL b. INT c. REAL
27. 什么类型数据储存在如下结构体数据类型中？
    a. Control b. Counter c. Timer
28. 在 ControlLogix 控制器中，定义应用在数据中的“数组”。
29. 比较数据在 SLC 处理器和在 ControlLogix 控制器中存储方式的异同。 125

习题

1. 基于典型的5位寻址方式，给下表中的每个输入和输出分配地址。

输入				输出			
设备	端子号	机架号	模块组编号	设备	端子号	机架号	模块组编号
a. 限位开关	1	1	3	a. 指示灯	10	1	0
b. 压力开关	2	1	3	b. 电机启动器	11	1	0
c. 按钮	3	1	3	c. 电磁线圈	12	1	0

2. 按照指出的问题修改如下程序：
   a. 问题：嵌套程序触点（图5-44）
   b. 问题：垂直程序触点（图5-45）
   c. 问题：有些逻辑被忽略（图5-46）
   d. 问题：太多的串联触点（仅允许4个）（图5-47） 126

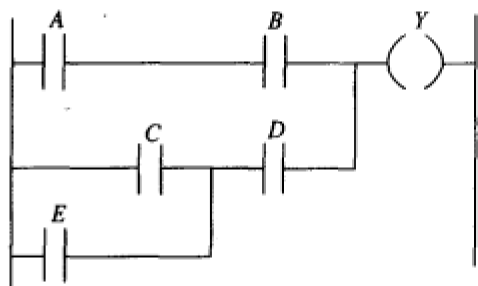


图　5-44

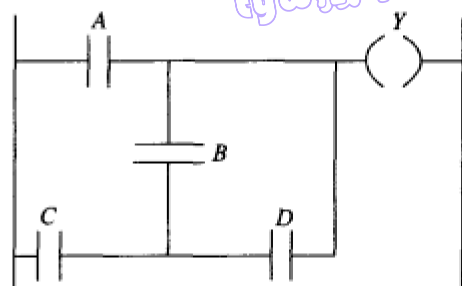


图　5-45

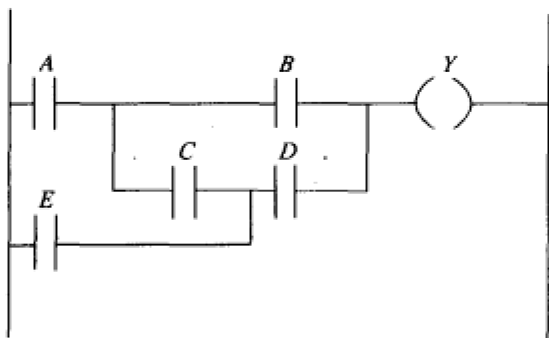


图　5-46

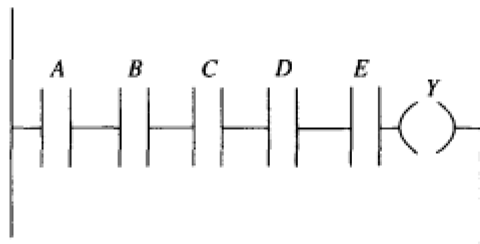


图　5-47

3. 画出与图 5-48 所示的硬接线电路所对应的梯形图程序：

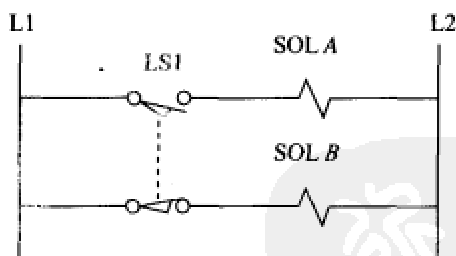


图　5-48

a. 用限位开关的一个常开（NO）触点连接 PLC 输入模块。
b. 用限位开关的一个常闭（NC）触点连接 PLC 输入模块。

4. 如要用 PLC 编程实现图 5-49 所示的硬接线电路，确定：
   a. 所有输入现场装置；
   b. 所有输出激励装置；
   c. 能用内部继电指令编程的所有装置。
5. 完成以下要求的任务，你选择什么指令来完成？（说出你的理由。）
   a. 当传送带电机反向运转时接通一个指示灯，现场输入设备是位于传送带启动继电器上的一系列触点。电机正向运转时，触点闭合；当电机反向运转时，触点断开。
   b. 当按钮按下时，它操作一个电磁线圈。输入现场设备是一个常开按钮。
   c. 当按下按钮时，电机停止运转，输入设备是一个常闭按钮。
   d. 当限位开关闭合时，它触发一个 ON 指令；输入设备是限位开关，当它闭合时在数据表位上储存“1”。

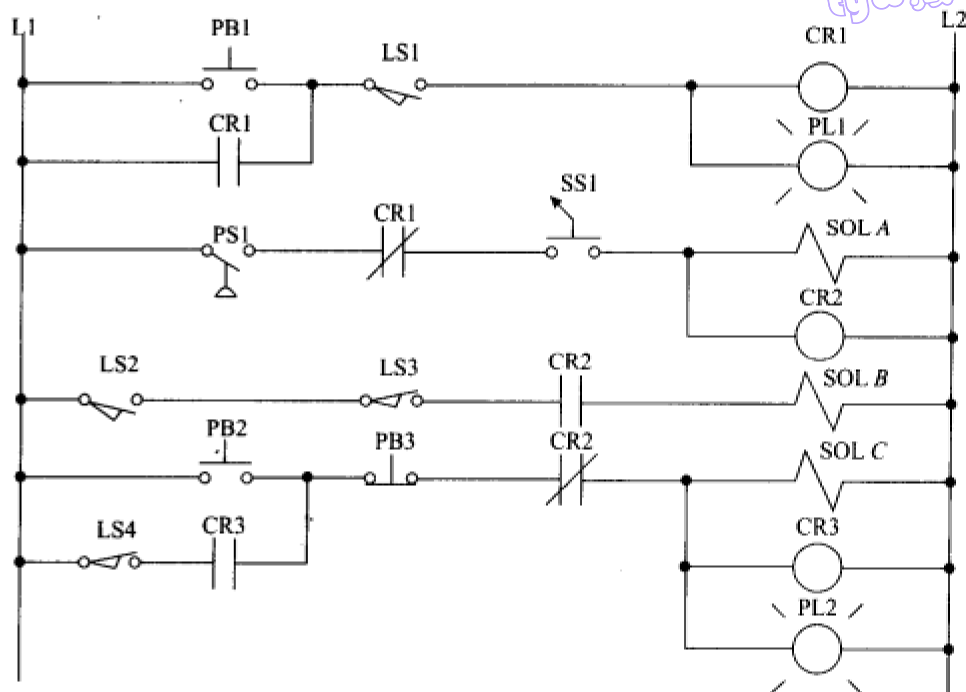


图 5-49

6. 写出满足下面需要操作的梯形图程序：（设输入 *A*、*B*、*C* 是常开的普通"乒乓"开关）

a. 当输入 *A* 闭合时，输出 *X* 和 *Y* 打开并保持 ON 态，直到输入 *A* 断开。

b. 当输入 *A* 闭合，而输入 *B* 或 *C* 断开时，输出 *Y* 处于 ON 状态，否则是 OFF 状态。

c. 当输入 *A* 断开或闭合时，输出 *Y* 处于 ON 状态。

d. 当输入 *A* 闭合时，输出 *X* 处于 ON 状态，输出 *Y* 处于 OFF 状态。

第6章 PLC 接线图与梯形图程序

学习目标：

- ❑ 识别电磁控制继电器、接触器和电机启动器各自的功能。
- ❑ 识别 PLC 装置中常用的开关。
- ❑ 解释 PLC 装置中常用传感器的工作原理。
- ❑ 解释 PLC 装置中常用输出控制设备的工作原理。
- ❑ 描述电磁控制锁存继电器和 PLC 中可编程锁存/解锁指令的操作。
- ❑ 比较顺序控制和组合控制的控制过程。
- ❑ 将基本继电器梯形图转换为 PLC 梯形图程序。
- ❑ 根据解释性的描述直接编写 PLC 程序。

为了便于理解，可以将梯形图程序和继电器电路进行比较。本章给出了传统继电器电路怎样转变为 PLC 梯形图程序的例子。通过本章可以了解到更多和 I/O 模块相连的现场设备。

6.1 电磁控制继电器

使用 PLC 的最初目的，是为了用一个可编程的固态交换系统来代替电磁式继电器。虽然在大多数场合下，PLC 能够代替继电器的逻辑控制，但电磁式继电器仍然作为现场开关 I/O 设备的辅助设备。设计 PLC 的目的是为了代替做出逻辑决策的小型控制继电器，而不是处理大电流或高电压。另外，了解了电磁式继电器的工作原理和术语对正确地将继电器电路图转换为梯形图程序非常重要。

继电器是一种磁性开关器件。它利用电磁作用来对触点进行控制。一个继电器通常只具有一个线圈，但是具有很多个触点。图 6-1 显示了典型的控制继电器的操作。在没有电流流过线圈时（失电），由于弹簧拉力作用，电枢保持在远离线圈铁心的位置；当线圈得电时，将产生电磁场。由于磁场的作用，引起电枢的物理动作。电枢动作将导致继电器触点断开或闭合。线圈和触点是相互隔离的，因此在正常状态下，在线圈和触点之间不存在电路。

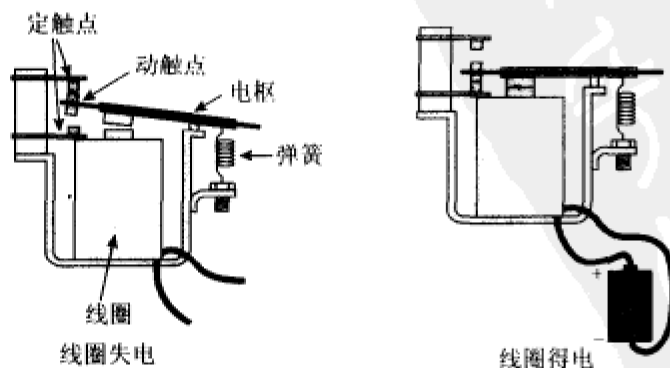
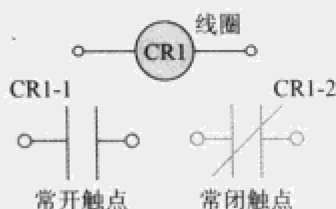


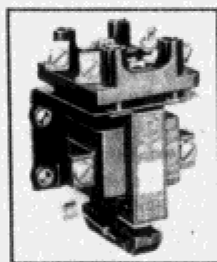
图 6-1 电磁控制继电器的运行

表示控制继电器的符号如图 6-2 所示。触点由一对平行的短线表示且通过相同的字母和编号（CR1）来辨别。图中显示了常开触点和常闭触点。常开触点的定义为当没有电流

流过线圈时触点处于断开状态，但是一旦线圈得电，触点立即闭合。常闭触点的定义为当线圈失电时处于闭合状态，但是一旦线圈得电，触点就立即断开。通常所画的每个触点都表示线圈为失电时的状态。



(a) 控制继电器符号



(b) 典型工业控制继电器（获得Allen-Bradley公司许可）

图 6-2　控制继电器

图 6-3 显示了一个典型的控制继电器用于控制两个指示灯的电路。开关打开，线圈 CR1 为失电状态。由于连接绿色指示灯的电路中线圈 CR1-2 与常闭触点相连，因此绿色指示灯发光。同时，由于连接红色指示灯的电路中线圈 CR1-1 与常开触点相连，因此红色指示灯熄灭。

129～130

由于开关闭合（如图 6-4 所示），线圈失电。常开触点 CR1-1 闭合使得红色指示灯亮。同时，常闭触点 CR1-2 断开使得绿色指示灯熄灭。

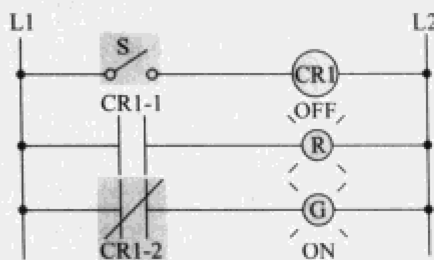


图 6-3　继电器电路 - 开关打开

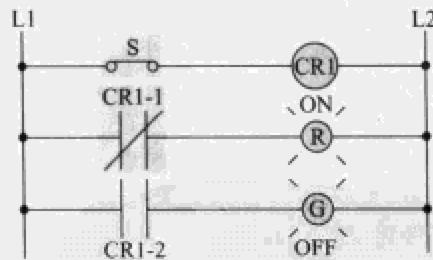


图 6-4　继电器电路 - 开关闭合

继电器通常只具有一个线圈，但是它可以有许多常开触点和常闭触点。控制继电器通常不必在大电流或高电压下进行工作。由于大多数线圈的公共电压等级为交流 120V，因此流过触点的电流等级通常在 5A 和 10A 之间。

6.2　接触器

接触器是用于处理超过控制继电器承受能力的重负载的特殊类型继电器。与继电器不同的是，接触器在不被破坏的情况下用于对电源电路的开通和关断。这样的负载包括照明器、加热器、变压器、电容和电动机，它们有些需要提供独立的过载保护，有些不需要，如图 6-5 所示。

控制继电器具有处理过电流和高电压的触点。PLC 具有能控制接触器的 I/O 口，但是不能直接控制大电流负载。图 6-6 显示了 PLC 和接触器连接控制水泵启/停的应用实例。输出模块和线圈串联，形成一小电流开关电路。接触器的触点和水泵的电动机串联，形成高电流开关电路。

131

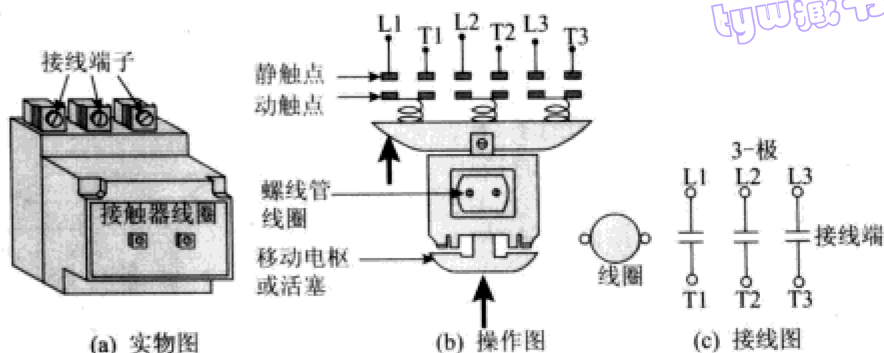


图 6-5　磁性接触器

利用磁性接触器来代替手动对设备进行控制的优点如下：

❑ 在处理大电流或高电压的场合，很难建立一个合适的手动装置。这样的装置的体积很大且难以操作。另一方面，用磁性接触器来处理大电流或高电压要相对容易。手动装置只需要对接触器的线圈进行控制。
❑ 操作员可以对接触器执行多重操作，接触器还可以设置连锁，防止故障和危险操作发生。
❑ 在一小时内需要多次重复操作的场合，如果使用接触器，将节省很多工作量。操作员只需简单地按下按钮，接触器将自动按照正确的顺序进行工作。
❑ 接触器可以由传感导向设备自动控制。导向设备的功率和外型有限制且不能直接用于处理大电流。
❑ 利用接触器可以处理高电压且可以将高电压同操作员完全隔离，因此提高了装备的安全性。操作员同样不会靠近大功率电弧，大功率电弧作为一个危险源，通常会对眼睛造成损伤。

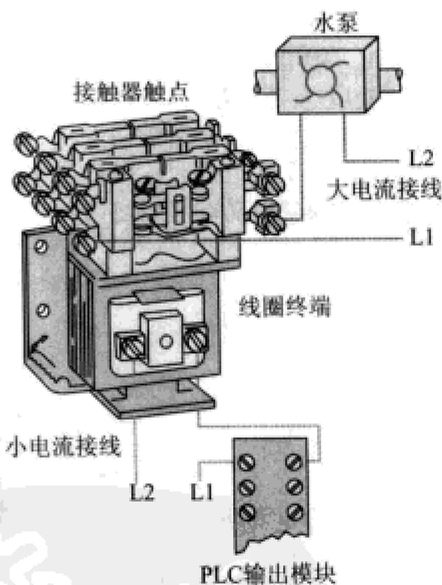


图 6-6　PLC 和接触器连接用于控制水泵的启停

❑ 通过使用接触器，控制设备可以被安装在远程点上。在靠近机器的地方只需要能够安装按钮的空间。在各个站之间只需铺设少量的光控线，就能通过多个不同的按钮控制同一个接触器。
❑ 通过使用接触器，使运用如 PLC 这样的设备实现自动和半自动控制成为可能。

6.3　电动启动器

电动启动器是一种专门为电动机提供电能的继电器。磁性启动器（如图 6-7 所示）是一种带有物理上和电气上相连的**过载继电器**的接触器。如果过载继电器检测到电动机过载，它将断开为启动器提供电源的电路。其实现方式是通过将发热器与向电动机提供电压的接触器串联。当电流流过发热器而使其发热时，热量直接传递给类似于双金属材料条的元件，元件加热后可以使机械式锁存器解扣。锁存器解扣使得与向电动机馈送电能的接触

器串联的触点断开。为了使电动机得到过载保护，发热器的特性应该和电动机相匹配。

磁性电动启动器是通过电磁原理工作的开关，它为大负荷的电动机提供了一种安全启动的方法。图6-8显示了一种典型的三相、磁性操作、交叉接线的交流启动器的接线图。按下启动器按钮，线圈M得电。此时，所有触点M闭合。与电动机串联的触点M闭合使得电流流入电动机。这些触点作为电源电路的一部分且必须能处理电动机的满载电流。当松开启动按钮，控制线圈电路中的触点M（与启动按钮并联）继续保持闭合状态。此触点作为控制电路的一部分且要求能处理使线圈得电的小电流。过载（OL）继电器用于提供电动机的过流保护。当检测到过流时，继电器常闭触点OL自动断开，因此线圈M失电且电动机停机。

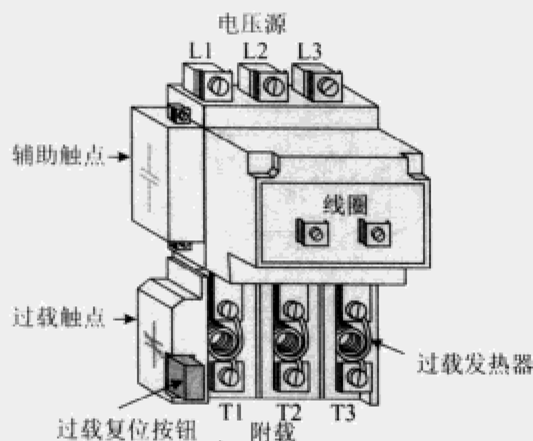


图6-7 磁性启动器

美国国家电气制造业协会（NEMA）规定了电动启动器的各种规格和等级。当PLC需要控制一个大电机时，PLC必须与一个启动器相连。启动器线圈所需要的电源必须在PLC输出模块的功率等级之内，如图6-9所示。

133

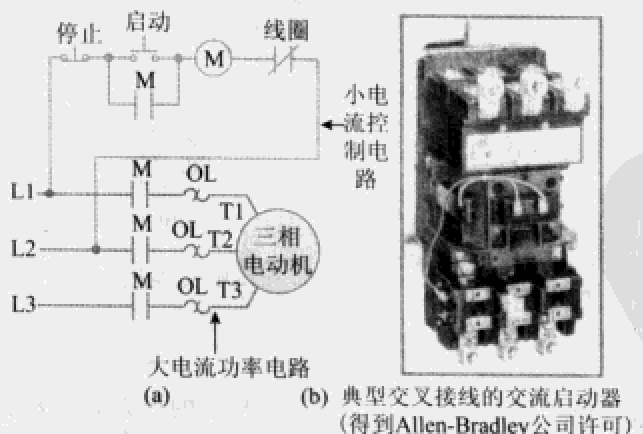


图6-8 交叉接线的交流启动器

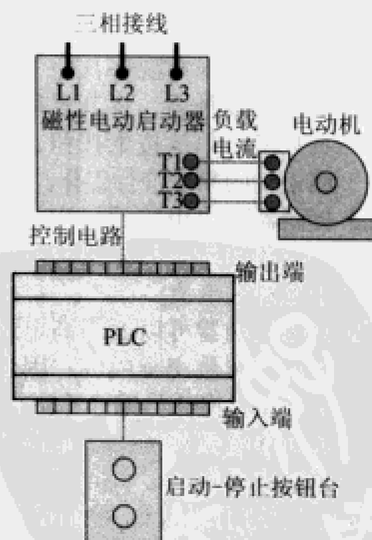


图6-9 PLC用于控制一个大电机时必须与启动器相连

6.4 手动开关

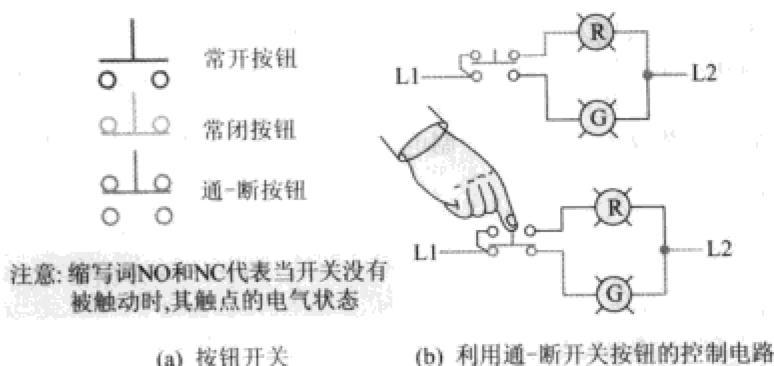
手动开关是通过手动进行控制。它包括转换开关、按钮开关、刀闸开关和选择开关。

按钮开关是工业中进行手动控制最常用的方式。图6-10显示了三种常用的按钮开关的符号。按下常开按钮将使电路接通，释放常开按钮将使电路恢复断开状态。按下常闭按钮将使电路断开，释放常开按钮将使电路恢复导通状态。通－断按钮用于互锁控制。在这

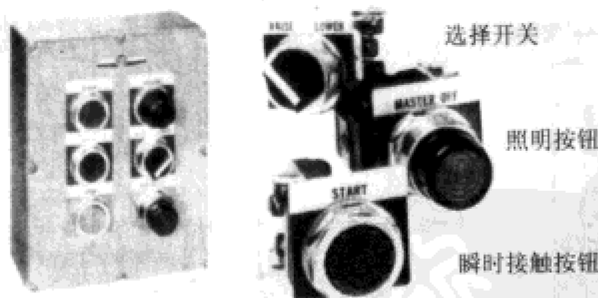
种按钮开关中，顶端的部分属于常闭，底端的部分属于常开。按下按钮时，底端的触点导通且顶端的触点断开。还有一种断开前导通按钮，按下按钮时，常开按钮在常闭触点断开前导通。 134

选择开关是另一种常用的手动操作开关。选择开关的位置状态是通过旋转旋钮来改变，而不是通过按下的方式。选择开关可以具有两个或多个可选状态，可以维持触点状态或者弹出返回以提供瞬时触点操作（如图 6-11 所示）。 135

双列直插式封装（DIP）开关是一种用于安装在印制电路板模块上的小型开关（如图 6-12 所示）。作为一种集成电路（IC）芯片，位于 DIP 开关底部的插线引脚或终端具有相同的大小和间隔。各个开关可以是触发器式开关、簧压摇杆开关或滑动触头开关。开关的设置几乎不会改动，且改动主要是在系统的安装或配置时发生。

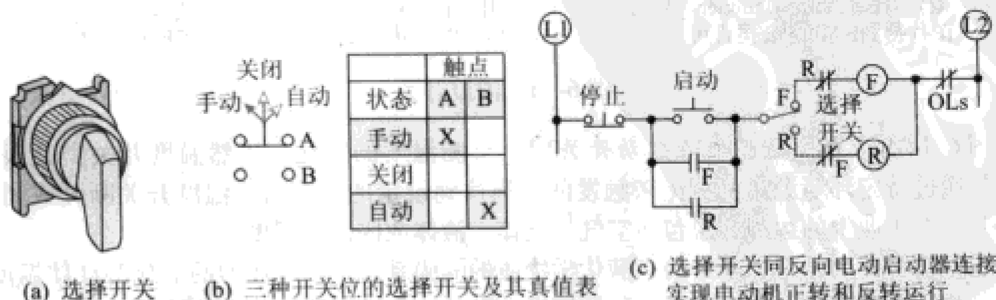


(a) 按钮开关　　(b) 利用通-断开关按钮的控制电路



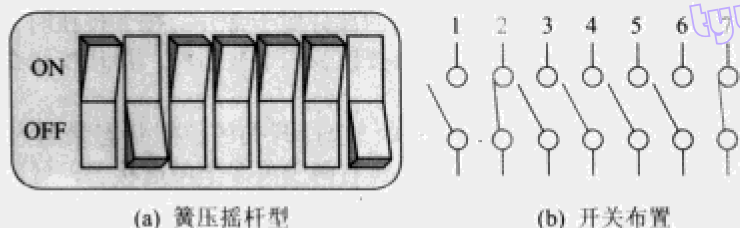
(c) 典型按钮控制台(得到Allen-Bradley公司许可)

图 6-10　各种类型的按钮符号和开关



(a) 选择开关　(b) 三种开关位的选择开关及其真值表　(c) 选择开关同反向电动启动器连接实现电动机正转和反转运行

图 6-11　选择开关



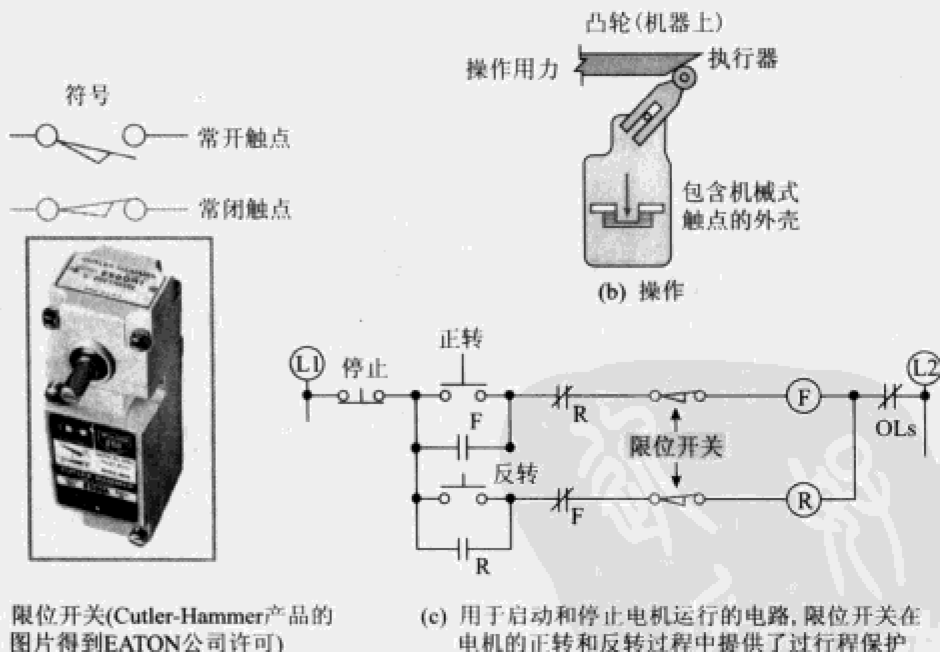
(a) 簧压摇杆型 (b) 开关布置

图 6-12 DIP 开关

6.5 机械动作开关

机械动作开关是通过压力、位置或温度等因素来进行自动控制的。图 6-13 显示的限位开关是一种常用的工业控制设备。只有到达预先设定的位置，限位开关才能动作，且限位开关通常是通过接触一些物体来被触发，如凸轮。这些设备可以代替操作员的操作。机械动作开关通常用于机器生产流程的控制电路，可以对电动机运行的启动、停止或反向进行控制。

136

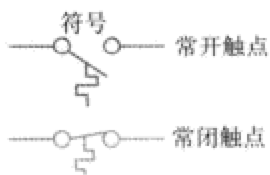


(a) 限位开关(Cutler-Hammer产品的图片得到EATON公司许可) (c) 用于启动和停止电机运行的电路，限位开关在电机的正转和反转过程中提供了过行程保护

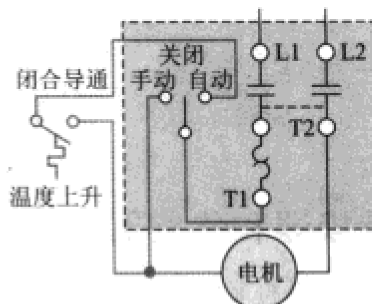
图 6-13 限位开关

图 6-14 显示的温度开关或热动开关用于检测温度的变化。虽然温度开关类型很多，但都是通过特定环境温度的变化来触发的。当达到设定的温度时，温度开关断开或闭合。温度开关在工业上的运用对象包括空气、气体、液体或固体的温度。

温度开关使用了一个密封的充满化学物质的隔膜系统。系统压力的变化与球体温度成比例。系统中反应温度的介质是一种易挥发的液体，它的蒸汽温度随着球状物温度的升高而提高。相反的，随着球体温度的降低，液体的蒸汽温度也下降。压力的变化通过一根毛细管传送到隔膜，在到达预先设置的温度时将对开关进行精确的操作。



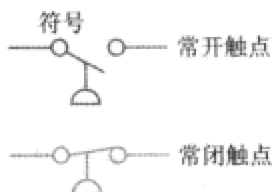


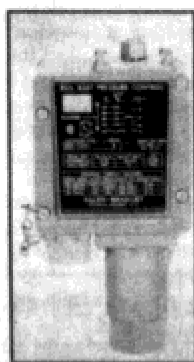


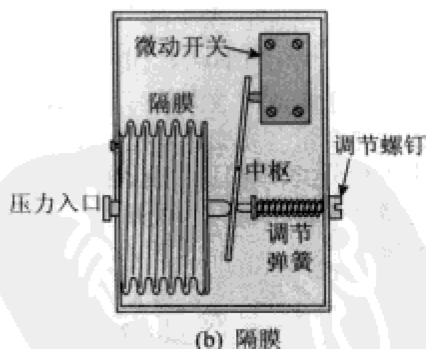
(a) 温度开关(图片由Allen-Bradley公司提供)　　(b) 用于自动控制电机的温度开关

图 6-14　温度开关

压力开关（如图 6-15 所示）用于控制液体或气体的压力。虽然压力开关有许多类型，但都是在达到预定的压力时才会触发其触点。压力开关可以是气动（空气）或液动（液体）开关。通常，压力开关的断开或闭合是通过隔膜或膜片来压迫一个微动开关来实现。 137







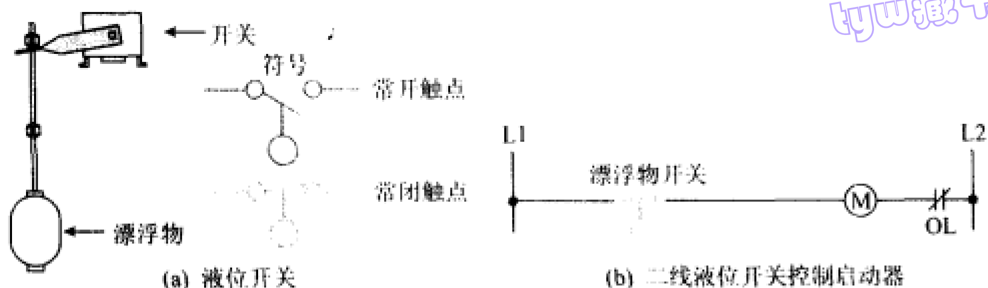
(b) 隔膜



(a) 压力开关(图片获得 Allen-Bradley公司许可)　　(c) 通过压力开关来操作启动器

图 6-15　压力开关

图 6-16 中的**液位开关**用于检测液体的液位。同液位开关机械式接触的漂浮物的上升或下降使得液位开关被触发；液位开关用于控制水泵电机的驱动，实现对容器的注水或放水。液位开关同样用于打开或闭合电磁阀门来控制液体流动。



(a) 液位开关　(b) 二线液位开关控制启动器

图6-16　液位开关

6.6 转换器和传感器

转换器是能够将能量从一种形式转换为另一种形式的设备。转换器可以分为两类：输入型转换器和输出型转换器（如图6-17所示）。电输入型转换器可以将如声音或光等非电信号转换为电信号。电输出型转换器工作顺序相反，将电信号转换为非电信号。

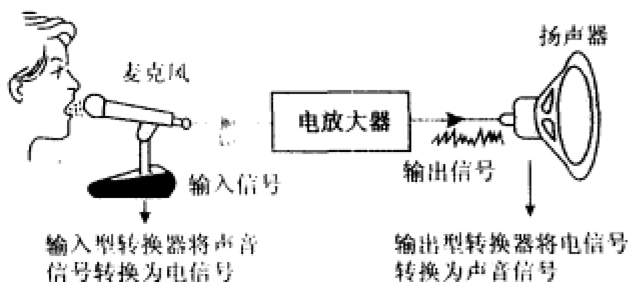
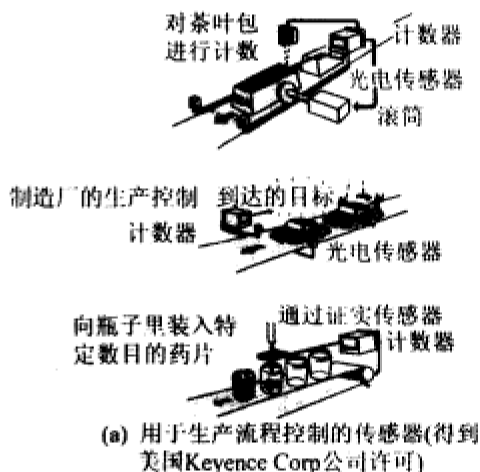


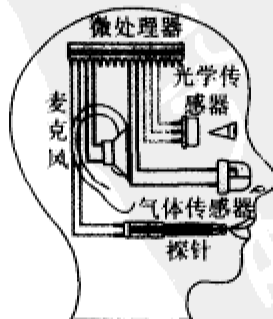
图6-17　电输入型转换器和电输出型转换器

传感器是用于检测时放大信号的转换器，并且通常用于测量信号。传感器能将机械变量、磁变量、热变量、光变量和化学变量转化为电压和电流。传感器通常是按照它们所测量的对象来进行分类，并且在现代生产流程控制中扮演非常重要的角色。如果把工业自动控制系统的微处理器比喻为“大脑”，则传感器就是它的“眼”、“耳”、“鼻”和“舌”（如图6-18所示）。

138



(a) 用于生产流程控制的传感器(得到美国Keyence Corp公司许可)



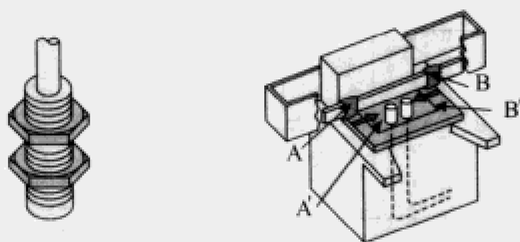
(b) 传感器比作微处理器“大脑”的“眼”、“耳”、“鼻”和“舌”

图6-18　传感器

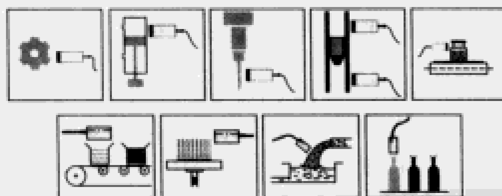
6.6.1 接近传感器

接近传感器或接近开关是在没有物理接触的情况下用于检测物体（通常称为目标）出现的指示设备（如图6-19所示）。它们为能够隔离工业环境中强烈振动、液体、化学物质和有害物质的固态电子设备。在下列情况下可以使用接近传感器：

- 在被检测的物体为很小、质量很轻或很软，使得机械开关难以操作的情况下。
- 需要快速反应且高速动作，例如在计数或喷射控制过程中。
- 物体隔着非金属障碍被检测，例如玻璃、橡胶和纸盒。
- 在恶劣环境下需要提高绝缘性，预防机械开关进行不恰当的操作。
- 需要长寿命和可靠的运行。
- 快速电子控制系统需要自由阶跃输入信号。



(a) 柱型外表　(b) 感应式接近传感器——典型的机器运用工具



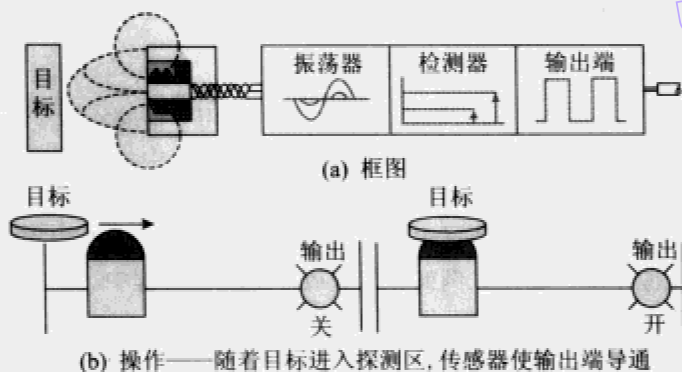
(c) 接近开关的应用(得到Rechmer Electronics公司许可)

图6-19　接近传感器

感应式接近传感器由一个金属物体触发。感应式传感器通常用于工业机器中。典型的应用如图6-19b所示。接近传感器（A'和B'）用于探测按照箭头方向移动的目标物体A和B。当A到达A'，机器运动反向；当B到达B'，机器再次运动反向。

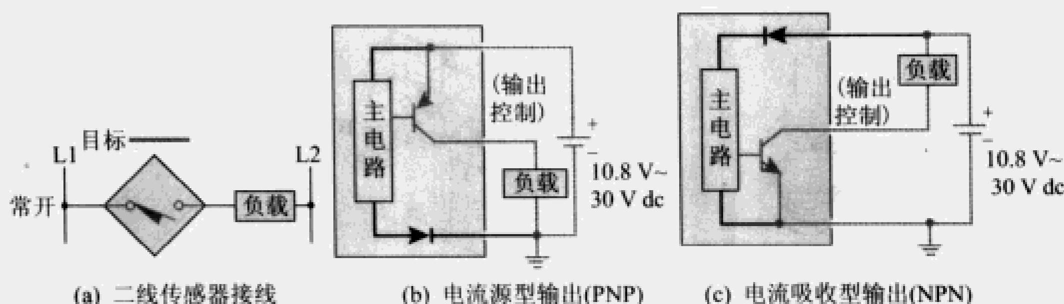
感应式传感器由线圈、振荡器、检测器电路和物理输出端（如图6-20所示）。当向传感器提供能量时，振荡器就开始工作产生一个高频域。此时在高频域内应该不存在任何导体。当一个金属物体进入高频域内，传感器将感应到物体表面的涡流。这些电流将导致振荡电路的能量损耗，进而引起振荡器一个较小的振幅。检测电路识别出振幅发生了特殊变化且产生一个输出信号，输出信号将使物理输出端变为导通或关断。当金属物体离开探测区域，振荡器恢复以前的振幅，使传感器返回到正常状态。

接近传感器的接线和启动的方式与传感器的类型和应用有关（如图6-21所示）。在电流源型输出或PNP晶体管输出条件下，负载应接在传感器和地之间。电流经过传感器流向负载，然后流入大地（射极打开）。电流源吸收型输出或NPN晶体管输出条件下，负载应接在传感器和电源正极之间。电流经过负载流入传感器，最后流向大地（集电极打开）。



(b) 操作——随着目标进入探测区，传感器使输出端导通

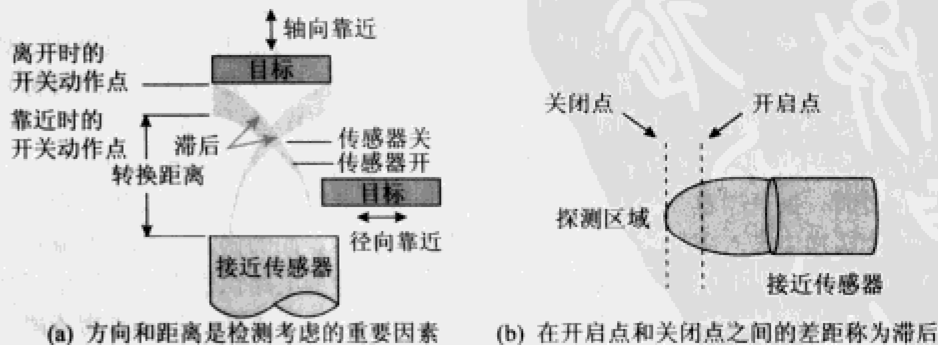
图 6-20　感应式接近传感器



(a) 二线传感器接线　(b) 电流源型输出(PNP)　(c) 电流吸收型输出(NPN)

图 6-21　接近传感器的接线

滞后表示的是两个工作点之间的距离。其中一个工作点是当被探测目标靠近接近传感器时的动作点；另一个是目标离开传感器时的释放点（如图 6-22 所示）。物体必须更靠近传感器使得其打开而不是关闭。如果物体朝着传感器移动，它将朝接近点运动。一旦传感器启动，它将一直持续到目标移动到释放点为止。当承受冲击和振荡、目标慢速移动或微小扰动时，例如电子噪声和温度漂移，需要用一个滞后来保持传感器的状态。



(a) 方向和距离是检测考虑的重要因素　(b) 在开启点和关闭点之间的差距称为滞后

图 6-22　滞后

作为物理开关动作输出的结果，即使当输出端已经关闭的情况下，也会有一个小的漏电流流过传感器。同样的，当传感器处于工作状态时，会在其输出终端产生一个小的压降。为了正确的运行，接近传感器应当连续带电。图 6-23 说明了用一个泄漏电阻来使得足够大的电流流过传感器，使得传感器正常运行，但是对开启 PLC 的输入端来说电流还不够。

140

电容式接近传感器是通过导体或非导体来触发的探测设备。电容式接近传感器的工作原理同样是基于振荡器的。然而不使用线圈，电容式传感器的激活面由两个金属电极构成——如同一个“断开的”电容。电极（如图6-24a所示）位于高频振荡器反馈环路中，高频振荡器在“没有当前目标”时为非活动状态。随着目标靠近传感器的表面，它将进入由两个电极形成的静电场内。此时将产生耦合电容，且电路发生振荡。振幅通过评价电路来测量，评价电路将产生一个信号使得物理输出端断开或导通。

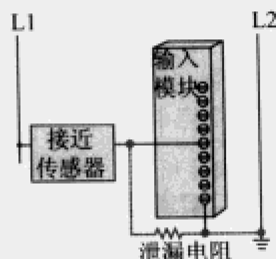
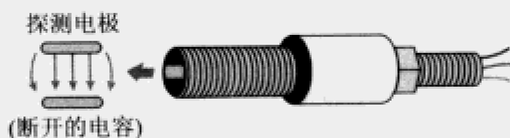
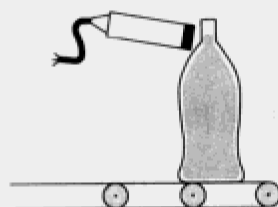


图6-23 接近传感器和PLC输入模块的接线

典型应用如图6-24b所示。利用电容式接近传感器可以在玻璃或塑料容器的外部控制液体填充容器。在一些应用中，当底部传感器探测到容器为空时，就开始注入液体。当液面到达顶部传感器时，停止注入液体。



(a) 操作



(b) 典型运用

图6-24 电容式接近传感器

要启动传感器，需要传导性介质。电容式传感器的启动既可以通过传导性介质，也可以通过非传导性介质，如木头、塑料、液体、糖、面粉和小麦等。虽然电容式传感器有这个优点，但是也带来一些不利因素（同感应式传感器相比）。例如，感应式接近开关只能通过金属启动，而对湿度、粉尘、污渍等不敏感。然而，电容式接近开关能够被它所处环境中的任何污物所启动。在一般运用情况下，电容式接近开关一般作为感应式接近开关的补充而非真正的替代品。当没有金属导体来启动感应式传感器时，用电容式传感器来补充（例如机器加工木材和确定液体的液位或粉尘的密度）。

6.6.2 磁开关

磁开关（也称簧片开关）的触点由两个扁平触点构成，触点密封在封闭的充满保护气体的玻璃管内。随着永磁场的靠近，与一个触点相接触的普通触点被另一触点吸引，离开当前触点去接触另一触点。随着永磁场的离开，触点失去磁性且返回其初始状态（如图6-25所示）。磁性簧片开关通常是通过弹性释放。由于触点被密封，不会被灰尘、湿度和烟气所影响；因此它们的预期寿命非常高。

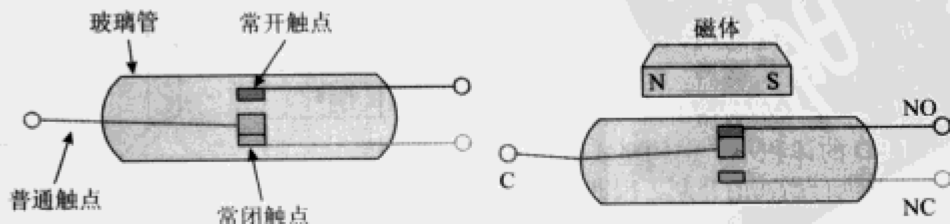


图6-25 磁开关（簧片开关）

永磁铁是簧片开关最常用的启动器。根据开关的动作要求，可以通过几种不同的布置方式启动永磁铁。最常用的一种布置方式为邻近运动、旋转和隔离方式（如图6-26所示）。也可以通过一个直流电磁铁来启动设备。当利用电磁铁操作时，通常称为簧片继电器。簧片继电器的操作更快且更可靠，和电动机械开关相比它们产生的电弧更小。然而，簧片继电器的当前处理性能具有一定的限制。

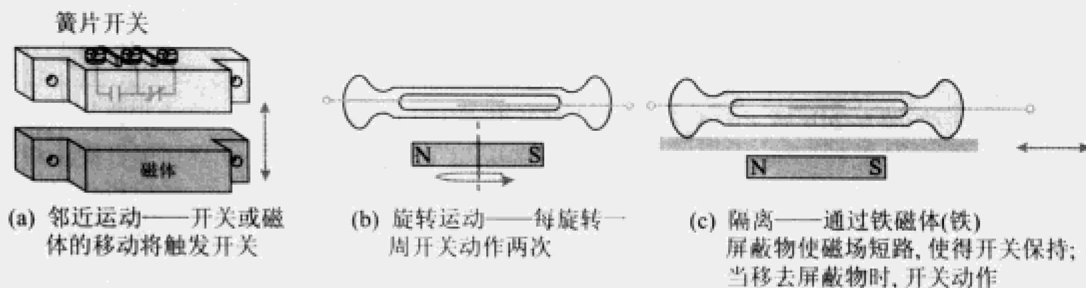


图6-26　簧片开关动作

6.6.3　光敏传感器

光电池或太阳能电池是常用的光敏器件设备，它能够将光能直接转换为电能（如图6-27所示）。现代的硅太阳电池就是利用了具有透明P层的PN结。透过透明层的光线使得P层和N层之间的电子发生移动，因此产生一小的直流电压。在阳光充足的情况下，每个电池能产生的典型输出电压大约是0.5V。

光敏电池（也称为光阻电池）是另一类比较流行的光转换器（如图6-28所示）。光能加在光敏电池上将使电池的电阻发生改变。硫化镉光电池是比较流行的一类光转换器。当器件的表面没有光照的时候，器件的阻值非常高。当有光照时，阻值就会下降到一个很小的值。

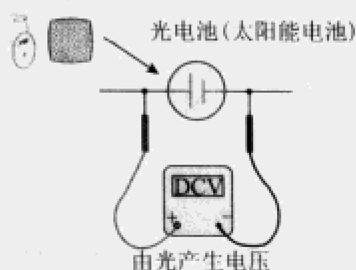


图6-27　光电池或太阳能电池

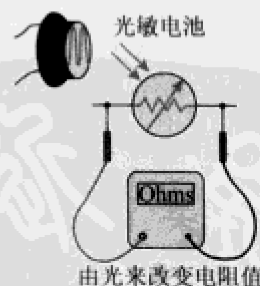


图6-28　光敏电池

大多数工业上的光电传感器使用发光二极管（LED）作为光源，用光敏晶体管来感测光是否存在（如图6-29所示）。在使用中，由LED发出的光照在光敏晶体管的输入端，通过晶体管光的数量发生了改变。通过光检测器，模拟输出端将按照可见光的量提供一定比例的输出信号。

由于LED的体积小、结构坚实且效率高，再加上它们能够进行高速的开断，因此通常被选作光电传感器的光源。LED通常处于不断的开断状态。与关断时间比，开通时间极短。LED通过脉冲进行

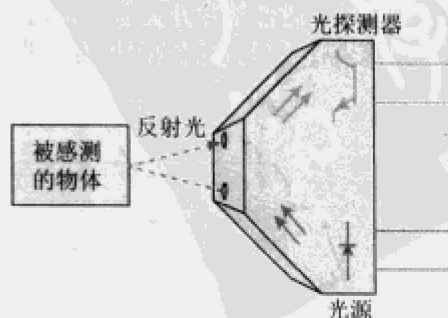


图6-29　光电传感器的运行

开断的原因有两个：为了使传感器不受周围环境光线的影响；为了增加 LED 的寿命。脉冲光线由光敏晶体管检测。光敏晶体管本质上是将脉冲光线从所有的外部环境的光线中分离出来。被选择的光源在人眼中通常是不可见的。选择特定波长是为了使传感器不受其他光线的影响。一些传感器允许探测不同的波长，称为色彩标识传感器，它们用于区分不同的颜色。

用于感测物体的光电传感器主要有两种类型。每个都从发光元件产生一个光束（可见的、红外的或激光）（如图 6-30 所示）。反射型光电传感器用于探测从目标反射回来的光束。透射型光电传感器用于测量由于目标通过光轴时所引起的光量的变化。典型的光电感测模式包括以下几种：

144

❑ **光感测**

当传感器接收到受调光束时，产生输出信号。

❑ **暗感测**

当传感器没有接收到受调光束时，产生输出信号。

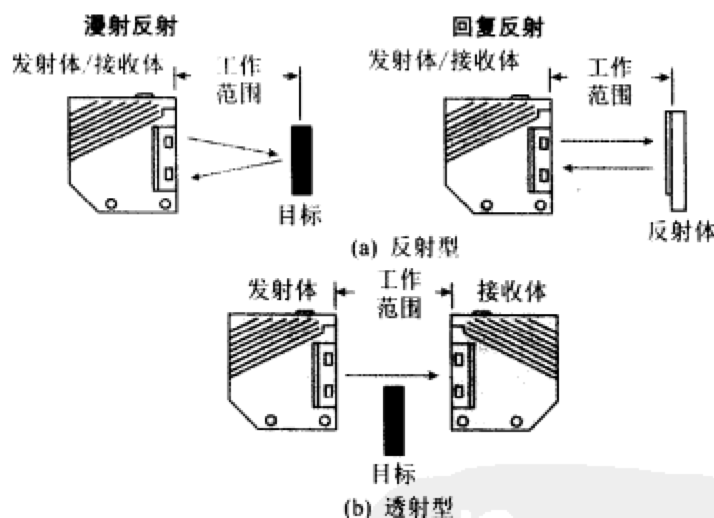


图 6-30　光电传感器的类型

图 6-31 显示了光电传感器的典型应用。这类传感器的特点包括如下：

❑ **无接触的探测**

无接触的探测可以避免目标或传感器的头部被损坏，确保更长的使用寿命和方便维修。由于会产生粒子，因此在清洁的空间里不能使用接触型的元件。

❑ **几乎能探测任何材料的目标**

探测是基于所接收的光量、光量的改变或反射光量的改变。此方法允许探测到不同材料的目标，如玻璃、金属、塑料、木头和液体。

❑ **长探测距离**

反射型光电传感器具有 1m 的探测距离，透射型光电传感器具有 10m 的探测距离。

❑ **快反应速度**

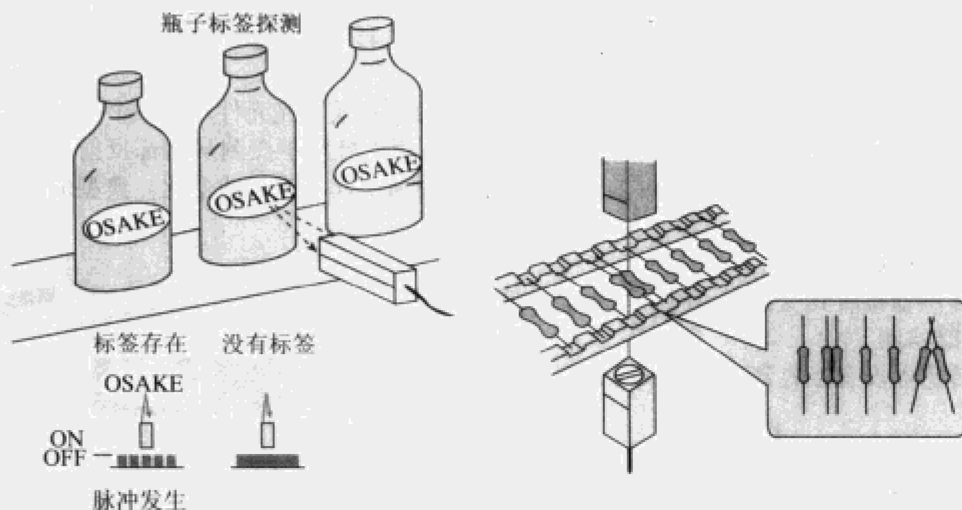
光电传感器的反应速度能够达到 50μs（1/20000s）。

❑ **颜色判别**

根据物体反射和吸收的颜色，传感器能探测到物体发出的光线，因此允许对颜色进行探测和区分。

□ **高探测准确性**

独特的光学系统和精密的电子电路允许对微小物体进行高准确度的定位和探测。



(a) 反射型光电传感器用于检测标签是否存在

(b) 透射型光电传感器端部分别位于电阻生产线的上下两端。生产线上的变化使激光束的强度发生改变，因此发出探测信号

图6-31 光电传感器的应用

条形码技术已经在工业上广泛运用，并且其应用的领域迅速扩大。条形码的使用很简单，通过使用条形码能使数据输入比手动输入数据更快，更准确。条形码系统包括三个基本部分：条形码符号、扫描仪和译码器。

145 条形码符号总共包含30个机器容易辨识的编码字符。字符通常都显示在条形码的上部或下部，这是为了在机器不能辨识符号时可以手动输入数据。在条形码符号两边的空白区，称为静态区，它同开始和结束字符一起用于使扫描仪区分数据的开始和停止（如图6-32所示）。

当前存在几种不同的条形码。在每种条形码中，数字、字母或字符都由特定数目的线条和空白形成。在美国，通用产品代码（UPC）作为零售食品包装袋上的标准条形码符号。在一个UPC符号（如图6-33所示）中包含所有的编码信息。UPC是严格的数字码，包含以下几部分：

- □ UPC的类型（1个字符）
- □ UPC的生产商，标识编号（5个字符）
- □ UPC的商品编号（5个字符）
- □ 用于检验读入数据准确性的校验数位（1个字符）



图6-32 条形码符号



图6-33 UPC条形码符号

条形码扫描仪是数据收集系统的眼睛。通过扫描仪中的光源照亮条形码符号；线条吸收光，空白反射光。光检测器通过代表条形码符号的电子信号形式对光信号进行采集。译码器接收从扫描仪传送过来的信号后，将数据转换为能表示条形码符号的字符型数据。虽然扫描仪和译码器的工作分为一组，但是根据实际的应用情况，可以把它们集成到一起或者分开工作（如图 6-34 所示）。

条形码模块也可用于 PLC。一个典型的应用是当盒子在传送带上移动时，通过条形码模块读取盒子上面的条形码，然后使用 PLC 将盒子转向合适的生产线（如图 6-35 所示）。

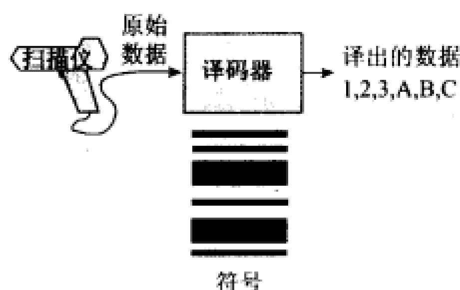


图 6-34 条形码扫描仪和译码器

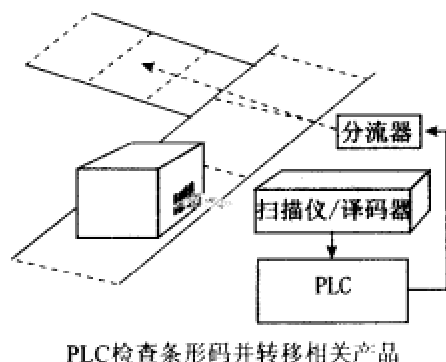


图 6-35 典型的条形码在 PLC 中的运用

6.6.4 超声波传感器

超声波传感器向目标物体发出声波，然后计算声波返回的时间。由于声音的速度是固定的，因此声波返回到传感器的时间直接和物体的距离或高度成正比。在图 6-36 中，返回的声波通过电转换，成为 4mA 到 20mA 的电流输出信号，能够向外部控制设备提供被监控的流动速度。通过超声波传感器可以对固体、液体、颗粒物体和纺织品等进行探测。声音从液面返回的速度和从固体物体表面返回的速度是相同的。纺织品和泡沫类物质能吸收声波，减少感测的范围。

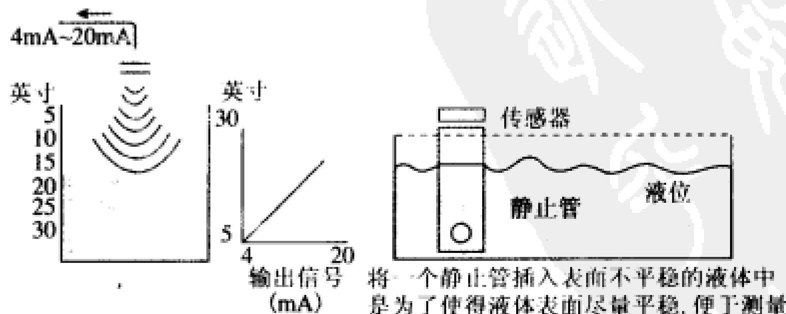
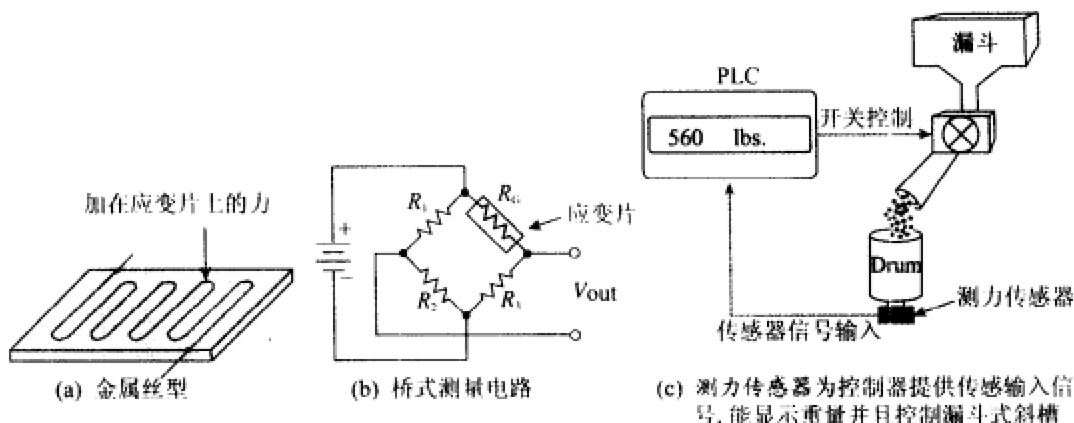


图 6-36 超声波传感器

6.6.5 应变/重量传感器

应变片转换器能将机械应力转换为电信号。应变片基于导体的电阻值随着导体的长度和横截面的变化而发生改变的工作原理（如图 6-37 所示）。加在应变片上的力使得应变片发生弯曲。应变片的弯曲使得其物理形状发生改变，从而导致其阻值发生变化。阻值的变化反馈给一个桥式电路，此桥式电路用于探测应变片的阻值所发生的微小变化。应变片测

[147] 力传感器通常由金属条和敏感应变片构成。随着测力传感器的负重变化，金属发生微小的伸长或缩短。应变片探测到金属微小的形变后，将其转换为变化的电压信号。有各种不同形状和大小的测力传感器，它们的敏感度范围从克一直到数百万磅。



(a) 金属丝型 (b) 桥式测量电路 (c) 测力传感器为控制器提供传感输入信号，能显示重量并且控制漏斗式斜槽

图 6-37 应变片

6.6.6 温度传感器

目前，常用的温度传感器主要有四种基本类型：热电偶传感器、电阻式温度检测器（RTD）、热敏电阻传感器和 IC 传感器。图 6-38 对这几种器件的特性进行了比较。

	热电偶传感器	电阻式温度检测器	热敏电阻传感器	IC传感器
	V 电压 / T 温度	R 阻值 / T 温度	R 阻值 / T 温度	V或者I 电压或电流 / T 温度
优点	• 自供电 • 简单 • 耐用 • 便宜 • 电压可变范围广 • 温度可变范围广	• 最稳定 • 最精确 • 线形度比热电偶传感器更好	• 高功率 • 快速 • 双金属丝测量	• 线形度最好 • 功率最高 • 便宜
缺点	• 非线性 • 承受电压低 • 需要参考点 • 最不稳定 • 最不敏感	• 昂贵 • 需要提供电源 • 阻值改变量小 • 电阻绝对值低 • 自动加热	• 非线性 • 温度范围受限 • 易碎 • 需要提供电源 • 自动发热	• $T<200°C$ • 需要提供电源 • 自动发热 • 结构上受限制

图 6-38 常用温度传感器

在工业应用中，热电偶是最常用的测量温度的元件。热电偶本质上是将一对不同导体的尾部焊接或熔融在一起形成“热点”或检测点，没有焊接在一起的点连接到“冷点”

或参考点。检测点和参考点之间温差的存在使得元件表现出热电偶的功能。当温差出现时，将产生一个小直流电压。由于热电偶的耐用性高和测量温度范围广，在工业上它被广泛用于检测和控制窑炉的温度（如图 6-39 所示）。

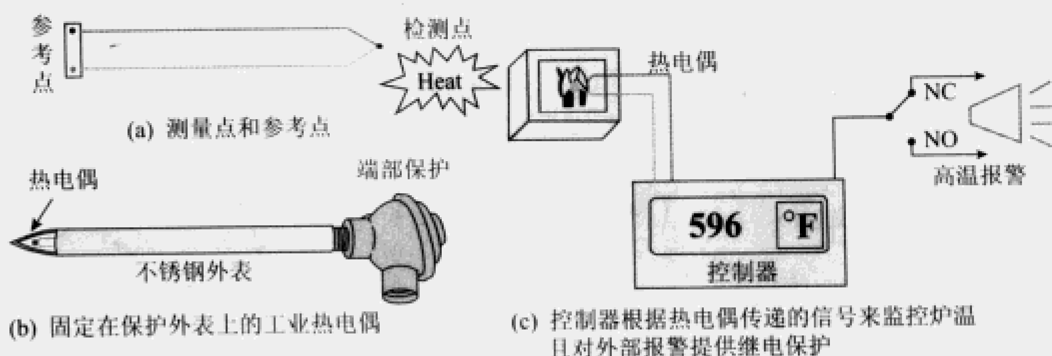


图 6-39 热电偶

6.6.7 流量传感器

许多工业过程都需要对液体的流量进行准确的检测。检测液体流量的方式不止一种，常用的方法是将液体的动能转换为其他可以检测的形式。通过将叶片和电位器相连可以实现简单的转换，通过将旋转的叶片和脉冲传感系统或测速计相连可以实现复杂的转换。如图 6-40a 所示的涡轮流量计，涡轮叶片的转速和液体流速成正比，并且在线圈中通过磁化产生一个感应电压脉冲。图 6-40b 显示了一个电磁式流量计，通过液体的流动产生感应电信号，且不会限制液体流动。线圈用于产生磁场。如果液体流过磁场，将会感应出电压并且被两个电极所感测。 148

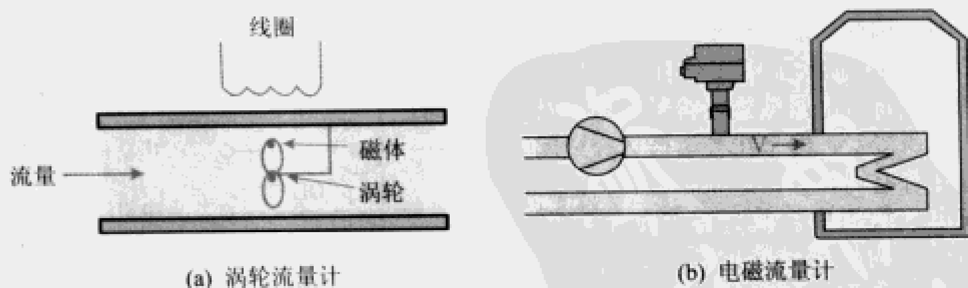
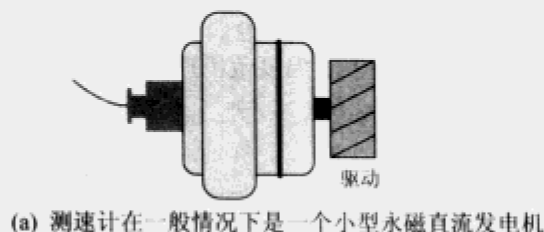


图 6-40 流量测量

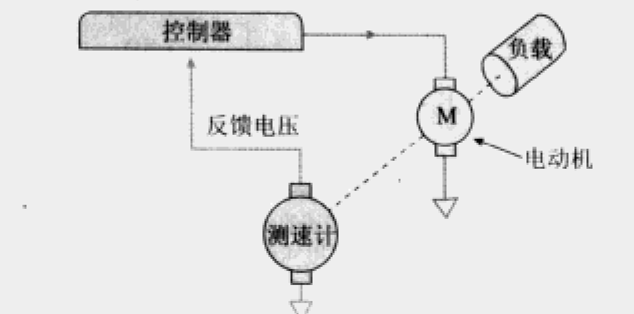
6.6.8 速度/转速传感器

发电机的输出电压随着发电机的转速改变而发生变化。通常用小型永磁直流发电机来代替测速计。当发电机开始运转时，它将产生和速度成一定比例的直流电压。被连接到电动机上的测速计通常用于在电机的速度控制中，向控制器提供一个和电动机的转速成比例的反馈电压（如图 6-41 所示）。 149

通常使用磁性（感应）传感器来测量轴的转速。磁体依附在轴上。每当磁体通过，绕制在磁体上的线圈将接收一个脉冲。通过测量脉冲的频率，可以确定轴的速度。常用的传感器的输出电压很小，在对其测量时需要放大（如图 6-42 所示）。

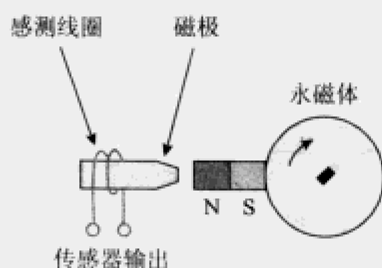


(a) 测速计在一般情况下是一个小型永磁直流发电机



(b) 测速计向控制器提供和电动机转速成比例的反馈电压信号

图 6-41　测速计



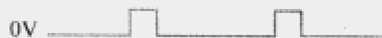


图 6-42　磁性传感器

6.7　输出控制装置

许多输出控制装置能够被控制器的输出模块操作，来控制传统的工业过程。这些装置包括指示灯、控制继电器、电动机启动器、报警器、散热器、螺线管、螺线管操作阀、小型电机和喇叭。在继电器电路和PLC输出端连接图中，用电气符号来表示这些装置。因此，能够识别这些符号非常重要。图6-43显示了用于各种各样输出装置的电气符号。虽然这些符号一般都被工业人员所接受，但是在各生产厂家之间仍然存在着差别。

在电气检测中，执行器是能够将电气信号转换为机械动作的任何设备。执行器的主要类型有继电器、螺线管和电动机。

螺线管是用于将电气信号或电流转换为线性机械动作的设备。如图6-44所示，螺线管由一个带有可移动铁芯的线圈构成。当线圈得电时，铁芯（有时称做电枢）被吸入线圈。螺线管产生的吸引力或排斥力的大小由铜线圈的匝数和流过线圈电流的大小所决定。

螺线管操作阀由两个基本功能部件构成：

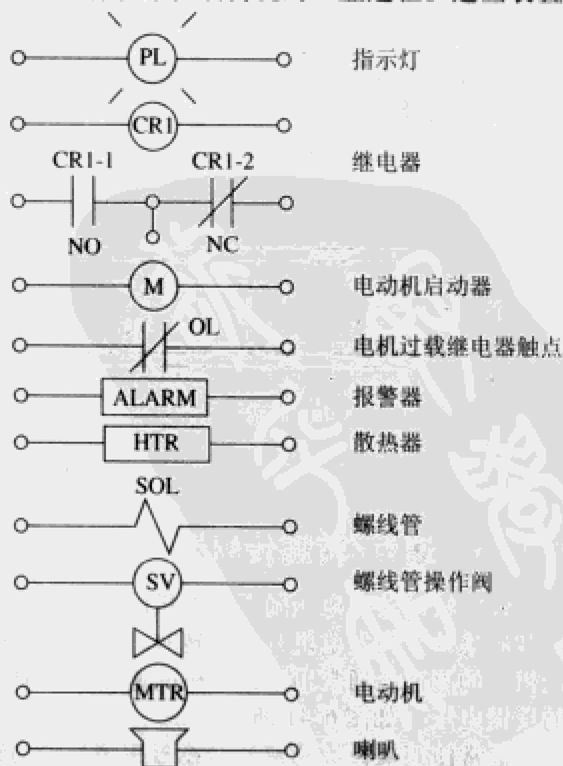
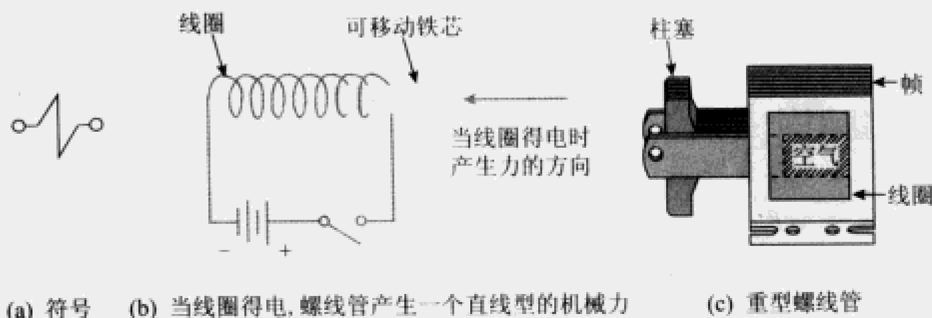


图 6-43　输出控制装置符号

150

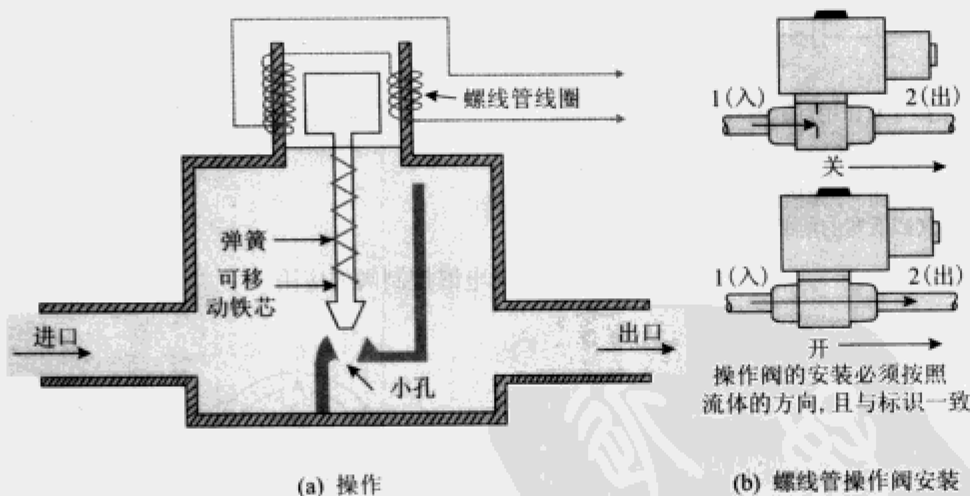
- 带有铁芯和柱塞的螺线管（电磁型）
- 带孔的操作阀，在孔中具有用于阻止流通或允许流通的一个盘或塞子

禁止或允许流体通过小孔取决于铁芯的运动和螺线管是否得电。当线圈得电时，铁芯被吸入螺线管线圈，使得操作阀打开。当电流消失时，弹簧使得操作阀返回到其原来的闭合状态。螺线管操作阀可用于液压系统（液体油）、气动系统（空气）或水流的控制（如图 6-45 所示）。



(a) 符号　(b) 当线圈得电，螺线管产生一个直线型的机械力　(c) 重型螺线管

图 6-44　螺线管



(a) 操作　(b) 螺线管操作阀安装

图 6-45　螺线管操作阀

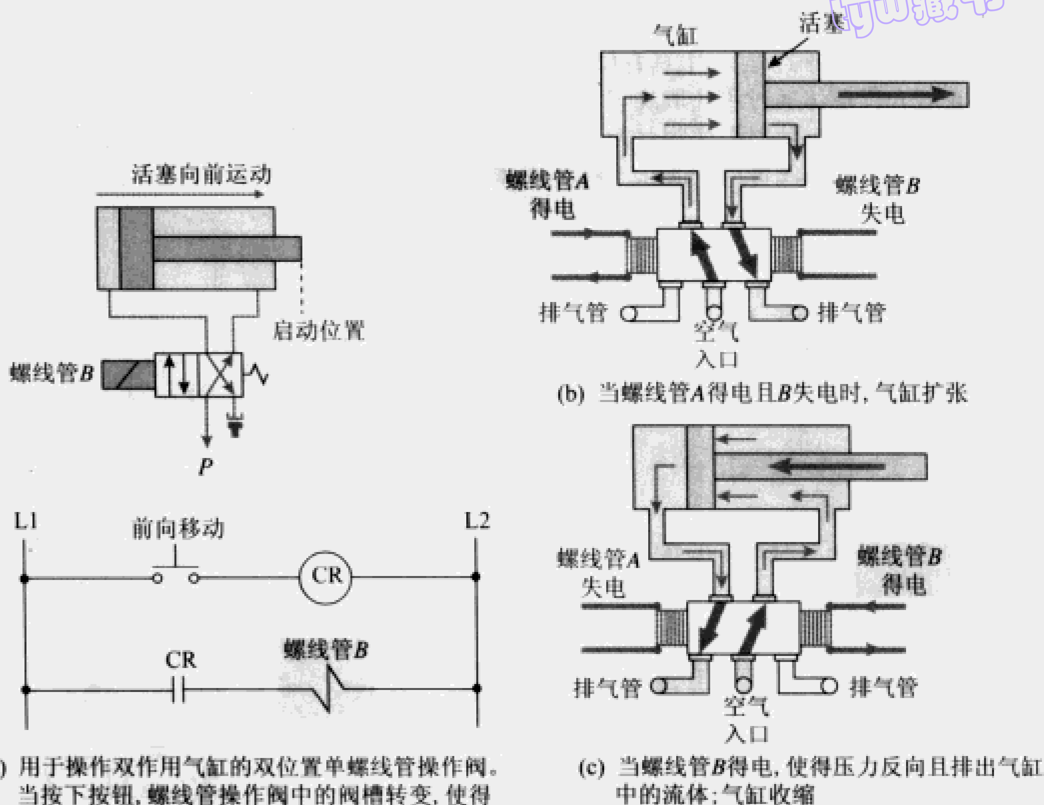
方向电磁阀（通常称作柱型阀）用于启动、停止和控制流体的流动方向。通过打开和关闭流通路径，操作阀可决定流体的方向。通常按照连接操作阀的数量和位置，对操作阀进行分类。图 6-46 说明了典型的方向电磁控制阀的应用。

步进电动机能将加在其上的电气脉冲转换为不连续的转子运动，此运动称为步进。每步一度的电动机运行一周将需要 360 个脉冲。微步进电动机每运行一周将达到几千步。步进电动机的等级通常按照每一周的步数来划分。它们通常处于低速和低转矩状态，且它们对运动提供精确的位置控制。

151

图 6-47 显示了步进电动机的典型 PLC 控制系统。步进电动机模块向能够控制步进电机的步进电机转换器提供脉冲。一旦 PLC 和步进电机模块连接，控制任务由 PLC 独立完成。步进电机模块将向转换器发出指令，且在此指令执行前不会接收其余指令。此模块的指令由 PLC 中的控制程序决定。

152

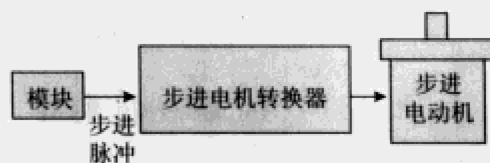


(a) 用于操作双作用气缸的双位置单螺线管操作阀。当按下按钮，螺线管操作阀中的阀槽转变，使得流体流向气缸活塞的背部，导致活塞向前移动。当松开按钮时，阀槽返回到原始位置，使得流体流向气缸活塞的前部，导致活塞反向移动

(b) 当螺线管A得电且B失电时，气缸扩张

(c) 当螺线管B得电，使得压力反向且排出气缸中的流体；气缸收缩

图 6-46　典型方向电磁控制阀的应用



(a) 典型的PLC步进电动机控制系统



(b) 驱动器每接收一个脉冲，步进电动机将运行一步。电脑以精确的或程序化等级提供期待数量的脉冲数，然后转换为距离和速度

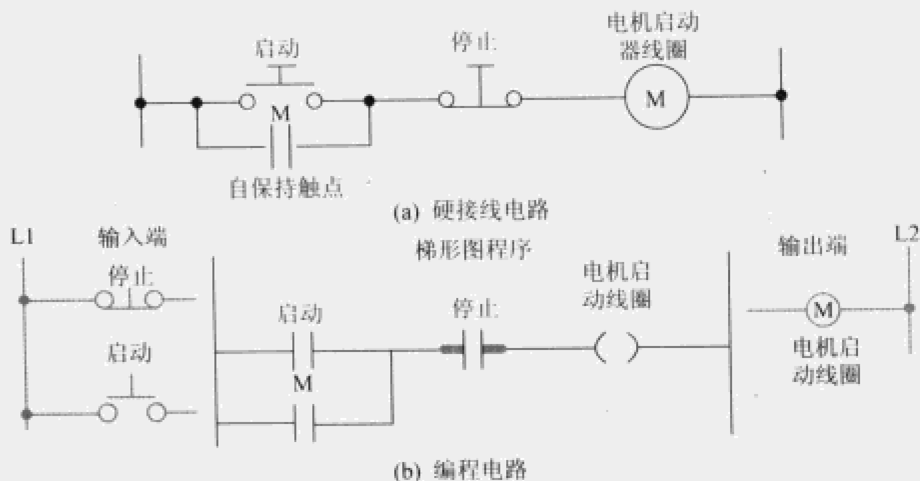
图 6-47　步进电动机

6.8　自保持电路

在继电器逻辑电路和 PLC 逻辑电路中，自保持电路或同步电路非常普遍。在瞬时开关已经按下或释放之后，自保持电路本质上是作为维持电流继续流动的一种方法。在多种类型的电路中，自保持触点通常和瞬时装置并联。

如图 6-48 显示的电动机启动/停止电路是自保持电路中的一个典型例子。硬接线的电路由一个常闭停止按钮和一个常开启动按钮串联构成。当松开启动按钮时，和启动按钮并

联的启动器的自保持辅助触点使得启动线圈保持得电状态。当此电路通过程序输入到PLC中时，由于启动按钮和停止按钮都必须闭合才能使得电机启动器运行，因此两个按钮都必须被检查其闭合的状态。



(b) 编程电路

图 6-48 自保持电路

6.9 自锁继电器

在自锁继电器的线圈失电时，其仍然能够保持继电器触点的闭合状态。当需要使触点保持打开和/或关断状态时，即使线圈只是暂时得电，需使用自锁继电器。图 6-49 为使用了两个线圈的自锁继电器。自锁线圈瞬时得电使继电器处于锁闭位置并且保持其状态。解锁线圈或释放线圈瞬时得电使机械锁扣解锁且继电器返回到非锁闭状态。

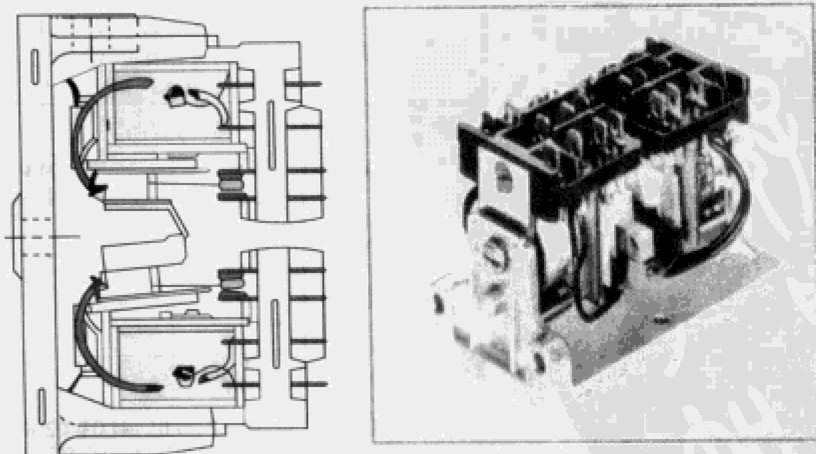


图 6-49 电磁自锁继电器（得到 Potter 和 Brumfield 公司许可）

图 6-50 显示了电磁自锁继电器的原理图。图中所示的继电器触点处于非锁闭位置。在此状态下，指示灯电路断开，灯不亮。当瞬时触发 ON 按钮时，自锁线圈得电，且使继电器置于锁闭状态。触点闭合，指示灯电路导通，因此指示灯亮。

注意，为了保持触点闭合且指示灯发光，继电器线圈没有必要处于持续得电状态。

使得指示灯熄灭的唯一方法是触发 OFF 按钮，这将使解锁线圈得电，然后触点返回到断开非锁闭状态。在失电的情况下，当电能恢复时，继电器将保持其原来的锁闭或非锁闭状态。

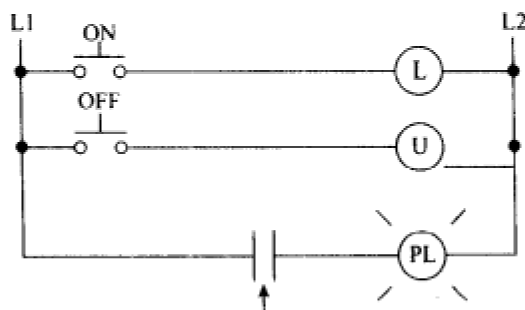
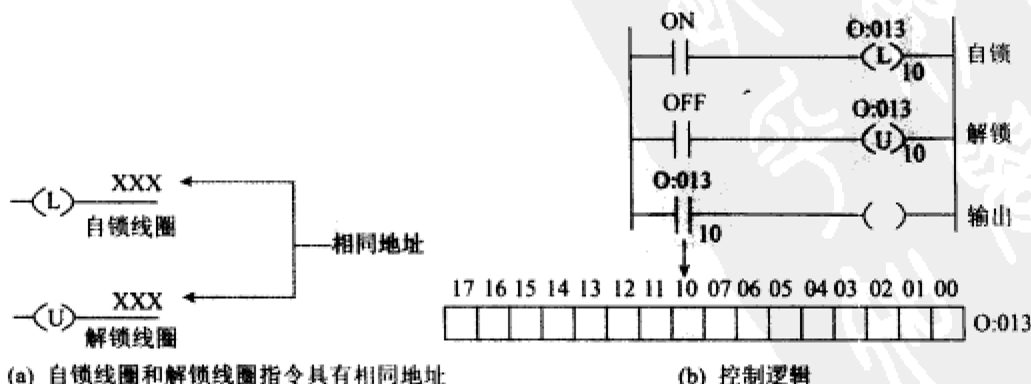


图 6-50 电磁自锁继电器原理图

可以对 PLC 进行编程实现电磁自锁继电器的功能。图 6-51 显示了输出自锁线圈指令和输出解锁线圈指令的作用。自锁（L）线圈和解锁（U）线圈具有相同的地址（O:013/10）。当瞬时触发 ON 按钮，自锁梯级状态变为真，且自锁状态位（10）被置 1，因此产生输出信号。当自锁梯级的逻辑连续性消失时，状态位将保持为 1。当解锁梯级状态变为真（触发 OFF 按钮），状态位（10）被复位到 0，因此指示灯熄灭。

154

指　令	名　称	符　号	描　述
OTL	输出自锁	-(L)-	当梯级状态变为真时，OTL 使状态位置 1，且当梯级连续性消失或电能循环出现时能保持其状态
OTU	输出解锁	-(U)-	当梯级变为真时，OTU 使状态位置 0 且保持



(a) 自锁线圈和解锁线圈指令具有相同地址　　(b) 控制逻辑

图 6-51 输出自锁和输出解锁指令

输出自锁指令是具有位级地址的输出指令。当指令为真时，它将在映像文件中对位进行设置。由于当自锁指令变为假时，位状态仍然保持，因此输出自锁指令属于保持型指

令。在大多数应用场合，自锁指令和解锁指令一起使用。输出解锁指令同样是具有位级地址的输出指令。当指令为真时，它将在映像文件中对位进行复位。由于当解锁指令变为假时，位状态不变，因此输出解锁指令同样是保持型指令。

图6-52所示的程序说明了输出自锁指令和输出解锁指令的运用。程序的功能是通过控制排水泵的开或闭进而对容器中的水位进行控制。操作如下：

❑ 关闭位

如果水泵处于运行状态，将停止；如果水泵处于停止状态，将不会启动。

❑ 手动模式

如果容器中的水位处于低水位以上的任何位置，水泵将被启动。

155

❑ 自动模式

- 如果容器中的水位到达高位点，水泵将启动使容器排水来降低水位。
- 当水位到达低位点，水泵停机。

❑ 状态显示指示灯

- 水泵运行指示灯（绿）
- 低水位状态指示灯（红）
- 高水位状态指示灯（黄）

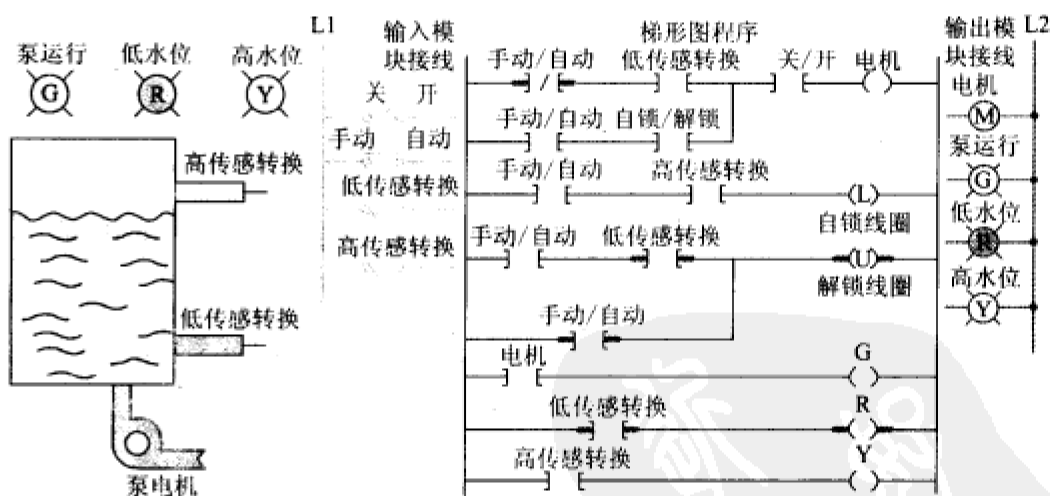


图6-52 容器中水位控制程序

6.10 继电器原理图向PLC梯形图程序的转换

根据继电器原理图设计PLC程序的最好方法，是先理解继电器原理图中每个梯级的工作原理。通过对每个继电器梯形图梯级的理解，可以生成一个等价的PLC梯级。这个过程需要获得继电器原理图、所用各种输入设备和输出设备的文档和操作过程流程图。

大多数的工业过程需要完成几个操作才能产生一个要求的输出。产品的制造、加工、装配、包装、抛光或传送需要各工作之间相互的精确协调。大多数工业控制处理器使用顺序控制方式。在根据特定的指令需要执行一定操作的流程中，需要使用顺序控制。图6-53显示了苏打水装瓶的部分流程。在装瓶和加盖操作中，任务是首先装瓶，然后加盖。任务的执行必须按照一个恰当的顺序。很明显，装瓶不能在加盖之后完成。因此，该流程需要顺序控制。

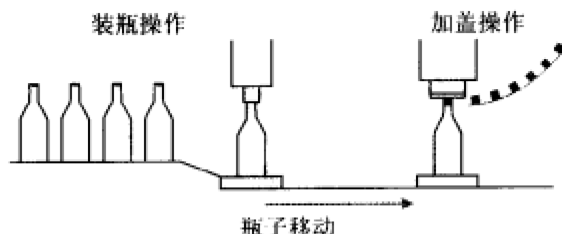


图 6-53 顺序控制流程

在不需要考虑被执行顺序的情况下，需要综合控制来执行特定的操作。图 6-54 显示了同样是苏打水装瓶流程的另一部分。在这部分中，任务是将标签 1 和标签 2 贴在瓶上。执行任务的顺序无关紧要。然而，实际上许多在最有效的操作顺序下执行的工业流程在本质上没有内在顺序。

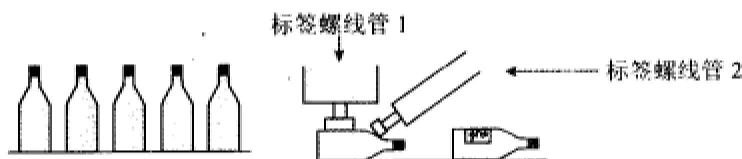


图 6-54 综合控制流程

自动控制使一个输出维持在设定值。维持熔炉的温度和容器中特定的液位就是好的例子。如果输出值和设定值之间存在偏差，就利用这个偏差对输出进行调整。这就需要从输出端引入反馈控制输入端。

156

如图 6-55 所示，通过参考简单任务，可以研究简单顺序流程的转换。图中为一个流程图和继电器梯形原理图的控制电路。任务顺序如下：

（1） 按下启动按钮。

（2） 电动机起动。

（3） 盒子运行到限位开关处，然后停止。

其他辅助功能包括：

157

（1） 无论何种原因，在盒子运行到限位开关之前，可以用紧急停止按钮停止电动机运行。

（2） 红色指示灯用于显示电动机停止。

（3） 绿色指示灯用于显示电动机运行。

如图 6-55 所示流程的控制任务可概括如下：

（1） 触发启动按钮；如果没有触发紧急停止按钮和限位开关，CR1 得电。

（2） 即使释放启动按钮，触点 CR1-1 也会闭合使得 CR1 自锁。

（3） 触点 CR1-2 打开，使得红色指示灯由发光变为熄灭。

（4） 触点 CR1-3 闭合，使得绿色指示灯由熄灭变为发光。

（5） 触点 CR1-4 闭合使得电动机启动线圈得电，电动机起动且使盒子朝着限位开关移动。

（6） 限位开关被触发，继电器线圈 CR1 失电。

（7） 触点 CR1-1 打开，自锁电路断开。

（8） 触点 CR1-2 闭合，使得红色指示灯由熄灭变为发光。

（9） 触点 CR1-3 打开，使得绿色指示灯由发光变为熄灭。

（10） 触点 CR1-4 打开，使得电动机起动线圈失电，电动机停机且顺序流程结束。

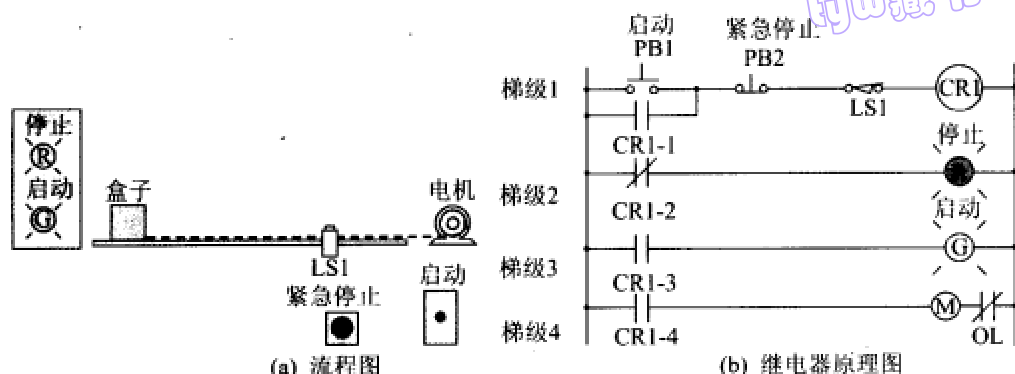
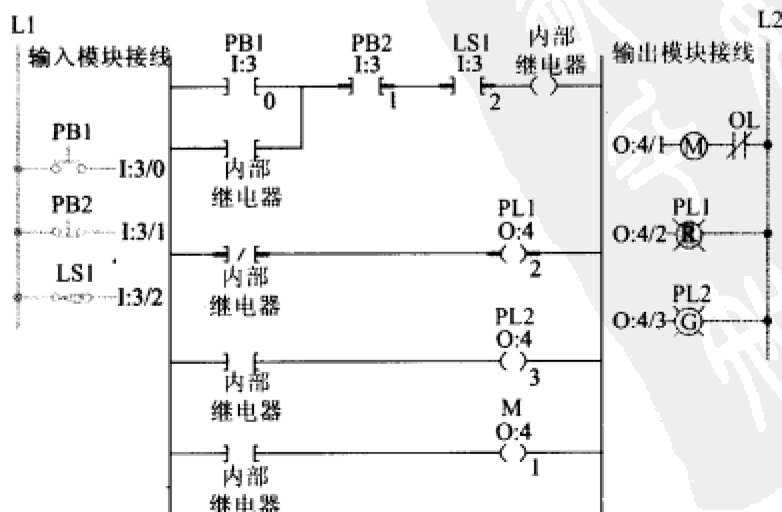


图 6-55　顺序流程

根据电路，应当列出一个 I/O 地址表。每个输入设备和输出设备可以用它们的地址来代表。这些地址将显示出 PLC 输入端和输入设备的连接情况，以及输出端驱动输出设备的情况。当然，地址编码将取决于所使用的 PLC 模块。图 6-56a 显示了在流程中使用 SLC-500 编址的典型 I/O 地址表。注意不必使用电磁控制继电器 CR1，因为它的功能已经被内部 PLC 控制继电器所替代。加在输入端和输出端上的符号是为了使程序更好地和现场设备相关联。

现场设备	逻辑地址	符号地址
启动按钮	I:3/0	PB1
紧急停止按钮	I:3/1	PB2
限位开关	I:3/2	LS1
电机启动器线圈	O:4/1	M
红色——停止指示灯	O:4/2	PL1
绿色——运行指示灯	O:4/3	PL2

(a) • 逻辑地址是在数据表中用于辨识地址配置的编号、字母和标点
• 由于符号地址和现实中的运用相关，因此它是能够替代逻辑地址的真实名称或代码。在数据表中，它是约定地址的真实名称



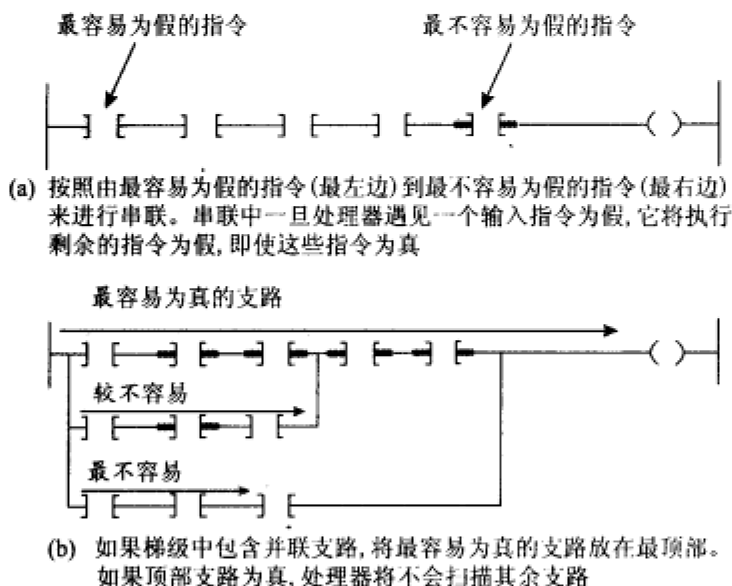
(b) 控制逻辑

图 6-56　程序

图 6-55b 中的继电器原理图的四个梯级可以转化为 PLC 编程语言的四个梯级，如图 6-56b所示。在梯级的转化过程中，必须了解每个梯级的功能。通过使用检查是否闭合指令对 PB1、PB2 和 LS1 编程来得到期望的逻辑控制动作。另外，内部继电器用来代替控制继电器 CR1。为了获得期望的控制逻辑，所有内部继电器触点通过使用与正常状态匹配（NO 或 NC）的 PLC 触点指令来编程。初始硬接线继电器 CR1 需要四个不同触点，然而内部继电器只使用一个可以检测任意次开关状态的触点。内部继电器设备的作用是 PLC 所独有的。

158

根据所给的控制流程有多种正确编写梯形图程序的方法。在一些情况下，恰当地布置程序可以优化程序存储区和程序扫描时间。图 6-57 显示了怎样优化程序。



159

图 6-57　编排指令使其优化

6.11　通过叙述式描述直接写梯形图程序

在多数情况，可以根据所描述的控制流程直接编写梯形图程序。编写程序的步骤如下：

- ❑ 明确控制流程。
- ❑ 画出流程的草图，包括所有控制顺序中需要执行的传感器和人工控制设备。
- ❑ 尽可能详细地列出操作的顺序步骤。
- ❑ 对 PLC 编写基本的梯形图程序。
- ❑ 考虑不同的方案，在处理顺序上根据需要可以不按照正常顺序并且进行调节。
- ❑ 考虑人工操作的安全性，并且根据需要进行调整。

例 6.1

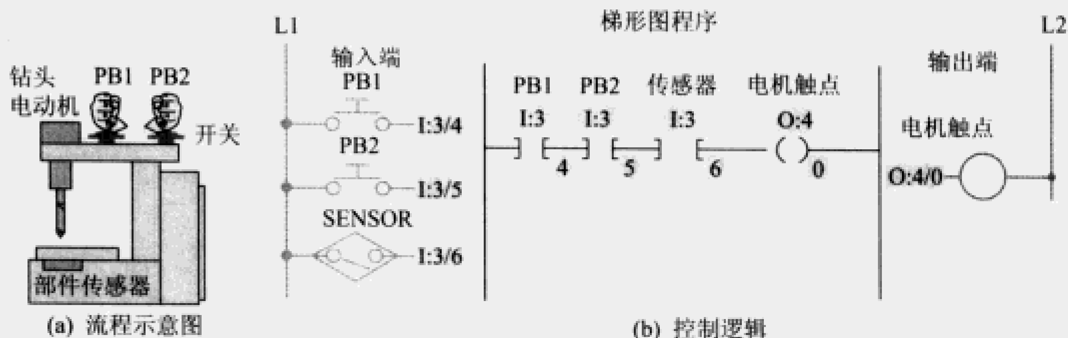
如果存在一个部件且操作员将手分别放在两个启动开关上，就可以进行简单的钻孔操作。预防措施将确保操作员的手和钻头保持一定的距离。

解决方法

- ❑ 图 6-58a 为流程的示意图。
- ❑ 操作顺序如下：

操作开关 1 和 2，必须激活部件传感器使得钻头电机运行。

❑ 图 6-58b 显示了流程所需的梯形逻辑。



(a) 流程示意图 (b) 控制逻辑

图 6-58 钻头操作流程

例 6.2

车库机动门可以自动执行开或关。现场设备包括如下：

- ❑ 用于控制上和下的换向电动机触点。
- ❑ 常闭低位限位开关。当门完全闭合时，常闭低位限位开关被触发，发出感测信号。
- ❑ 常闭高位限位开关。当门完全打开时，常闭高位限位开关被触发，发出感测信号。
- ❑ 使门向上运动的常开按钮。
- ❑ 使门向下运动的常开按钮。
- ❑ 使门停止运动的常闭按钮。
- ❑ 当门只开了一部分时显示门半开状态的红色指示灯。
- ❑ 当门全开时显示门打开状态的绿色指示灯。
- ❑ 当门全闭时显示门关闭状态的黄色指示灯。

解决方法

❑ 操作顺序如下：

- 当按下向上按钮时，上行电动机触点闭合，门向上运动，直到触动高位限位开关。
- 当按下向下按钮时，下行电动机触点闭合，门向下运动，直到触动低位限位开关。
- 当按下停止按钮时，电动机停机。电动机必须在它能改变方向之前被停下来。

❑ 图 6-59 显示了流程所需的梯形图逻辑。

例 6.3

连续填充操作需要传送带上的盒子能够自动定位，然后下料。

解决方法

❑ 图 6-60a 为流程的示意图。

❑ 操作顺序如下：

- 当按下启动按钮时，启动传送带。
- 当按下停止按钮时，使传送带停止运行。
- 当流程进行时，运行状态指示灯亮。
- 当流程停止时，预备状态指示灯亮。
- 当盒子右边沿首先被光电传感器探测到时，传送带停止运行且预备状态指示灯亮。
- 随着传送带停止运行且盒子静止不动，打开螺线管操作阀，向盒子下料。当位置传感器被触发时，停止下料。
- 当盒子装满时，装满状态指示灯亮。装满状态指示灯保持发光，直到盒子完全离开传感器的探测范围。

162

❑ 图6-60b显示了流程所要求的梯形图逻辑。

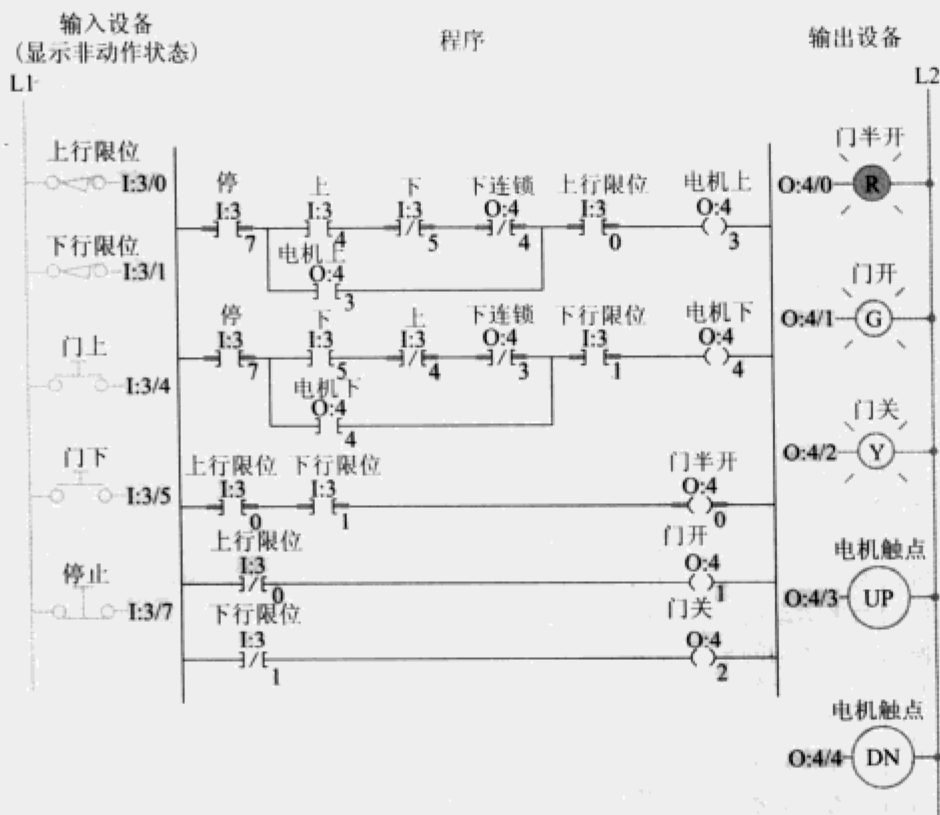


图6-59 机动门的PLC程序

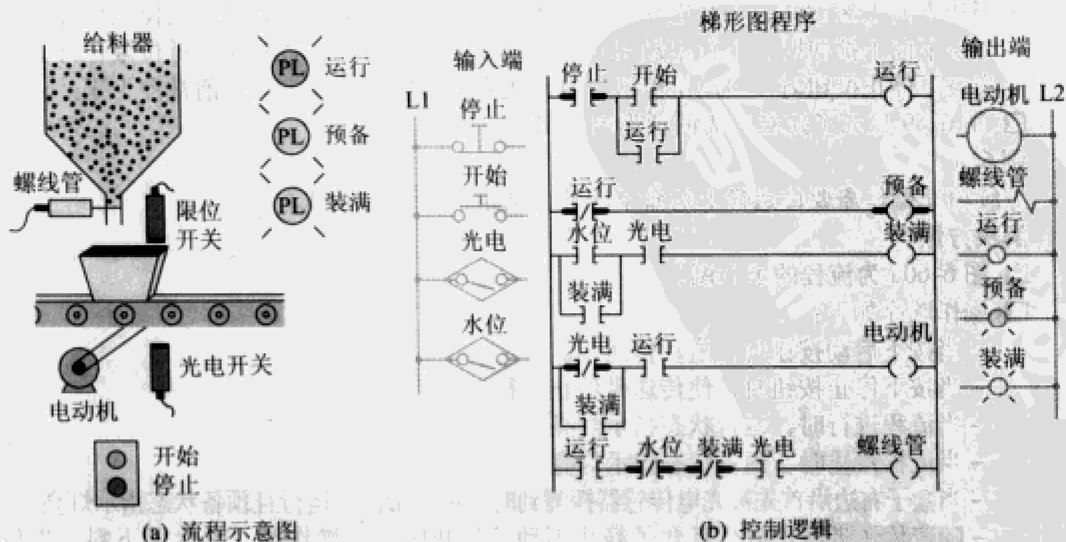


图6-60 连续下料操作

163

思考题

1. a. 解释电磁控制继电器的基本工作原理。
   b. 根据在电磁控制继电器中的应用，解释术语常开触点和常闭触点的含义。
2. 接触器和继电器在构造方式上有什么不同？
3. 接触器和磁性电动启动器最大的差别是什么？
4. a. 画出交叉接线交流启动器的原理图。
   b. 参考原理图，解释以下每部分的功能：
   （1）主触点 M　（2）控制触点 M　（3）启动器线圈 M　（4）过载继电器线圈
   （5）过载继电器触点
   c. 解释在启动器控制电路中和在启动器电源电路中对电流的要求有什么不同。
5. a. 比较以下几种类型开关的操作方式：
   （1）手动操作开关（2）机械操作开关　（3）接近开关
   b. 在描述开关触点时，缩写词 NO 和 NC 代表什么？
   c. 画出用于表示以下各种开关的电气符号：
   （1）NO 按钮　（2）NC 按钮　（3）断开－接通按钮
   （4）单极选择开关（5）NO 限位开关　（6）NC 温度开关
   （7）NO 压力开关（8）NC 液位开关　（9）NO 接近开关

164

6. 比较感应式传感器和电容式接近传感器的激活方式。
7. 比较光电池和光导电池的工作方式。
8. 用于光源和光感测设备的最常用的光电传感器是什么？
9. 比较反射型光电传感器和透射型光电传感器的工作方式。
10. 解释条形码扫描仪怎样读入条形码符号。
11. 解释超声波传感器的工作原理。
12. 解释应变片的工作原理。
13. 解释热电偶的工作原理。
14. 比较涡轮流量计和电磁流量计的工作方式。
15. 写出两种类型的速度传感器，并解释各自的基本工作方式。
16. 画出用于代表以下 PLC 输出控制设备的电气符号：
    a. 指示灯　b. 继电器　c. 电动启动器线圈　d. 过载继电器触点　e. 报警器　f. 散热器
    g. 螺线管　h. 螺线管操作阀　i. 电动机　j. 喇叭
17. 解释以下执行元件的功能：
    a. 螺线管　b. 螺线管操作阀　c. 螺线管方向操作阀　d. 步进电动机
18. 自保持电路是一种什么电路？
19. a. 画出简单电磁自锁继电器用于控制指示灯的电路图。
    b. 参考此电路，解释怎样实现指示灯的开和关。
    c. 在电路中，假设指示灯在发光时电路失电，当电源恢复时，指示灯是否发光？为什么？
20. 解释顺序控制和综合控制流程的区别。
21. 针对一个控制流程，以什么方式来排列程序能够使得效率更高？

165

习题

1. 设计并且画出一个传统的硬接线继电器电路，在按下 NC 按钮时，使之能实现以下要求：
   - ❑ 指示灯发光
   - ❑ 螺线管失电

- ❑ 启动电动机
- ❑ 喇叭报警

2. 设计并且画出一个传统的硬接线继电器电路，使之能利用两个断开－导通按钮实现以下功能：
   - ❑ 当按下按钮 PB1 时，接通指示灯 L1。
   - ❑ 当按下按钮 PB2 时，接通指示灯 L2。
   - ❑ 两个按钮电气连锁，实现 L1 和 L2 不能同时被接通。
3. 根据图 6-61 所示的梯形图程序，回答下列问题：

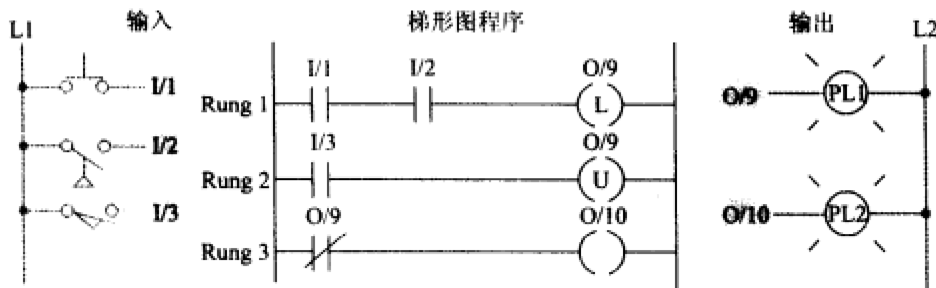


图 6-61

   a. 在什么条件下自锁梯级 1 为真？　　b. 在什么条件下解锁梯级 2 为真？
   c. 在什么条件下梯级 3 为真？
   d. 当指示灯 1 发光时，继电器的状态是什么（自锁还是解锁）？
   e. 当指示灯 2 发光时，继电器的状态是什么（自锁还是解锁）？
   f. 如果将交流电源从电路移去，然后又将其恢复，当电路电源恢复时，哪个指示灯将自动发光？
   g. 假设继电器处于自锁状态且所有输入都为假。要使继电器变为解锁状态，输入端应该发生什么变化？
   h. 如果地址 I/1、I/2 和 I/3 中检查闭合指令是否都为真，继电器将保持在什么状态（自锁还是解锁）？

166

4. 设计一个 PLC 程序，并且制作一个常用的 I/O 端连线图，通过执行梯形图程序能够正确地执行图 6-62 中的硬接线控制电路。

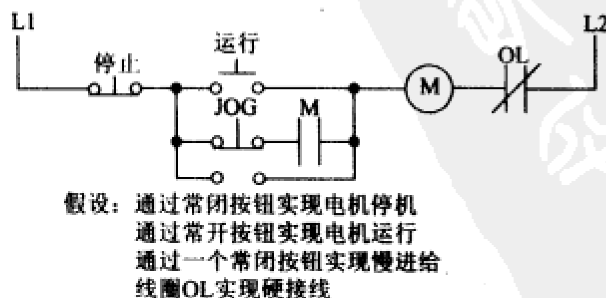


图 6-62

5. 设计一个 PLC 程序，并且制作一个常用的 I/O 端连线图，通过执行梯形图程序能够正确地执行图 6-63 中的硬接线控制电路。
6. 设计一个 PLC 程序，并且制作一个常用的 I/O 端连线图，通过执行梯形图程序能够正确地执行图 6-64 中的硬接线控制电路。

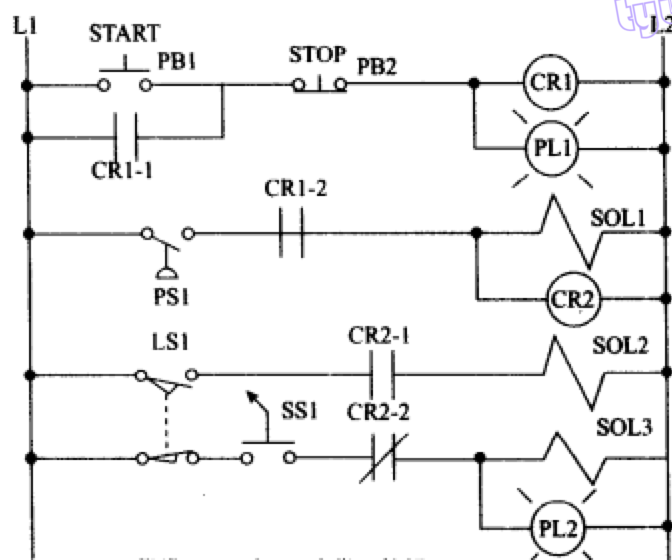


图　6-63

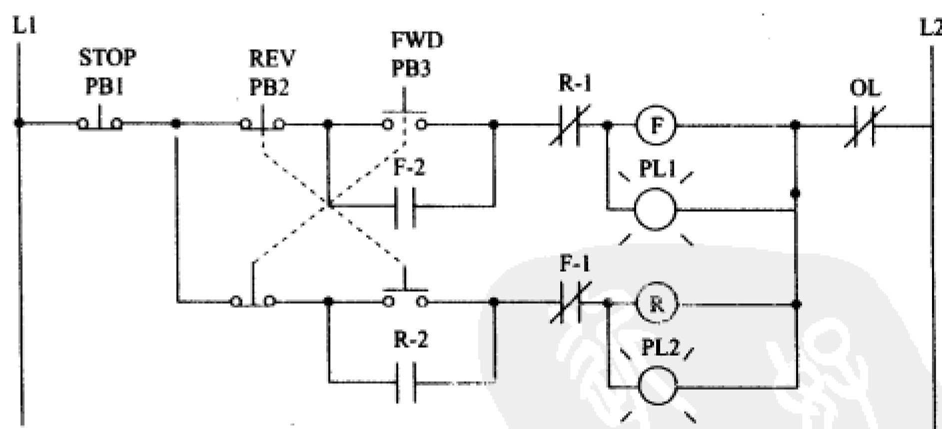


图　6-64

7. 设计一个 PLC 程序，并且制作一个常用的 I/O 端连线图，能够实现以下电机控制要求：
   - 通过三个启动/停止控制台中的任意一个按钮实现电机的启动和停止。
   - 每个启动/停止控制台包括一个常开启动按钮和一个常闭停止按钮。
   - 电动机过载继电器触点为硬接线。
8. 设计一个 PLC 程序，并且制作一个常用的 I/O 端连线图，能够实现以下电机控制要求：
   - 接线中含三个启动器。每个启动器的操作对应一个启动/停止按钮控制台。
   - 当按下主控台上的停止按钮，能够使所有启动器脱扣。
   - 通过编程使当任意一个启动器过载时，过载继电器触点动作使启动器停止工作。
   - 所有按钮都使用常开触点。

9. 某温度控制系统由4个温度调节装置组成，控制3个加热单元。每个温度调节装置分别设为50°F、60°F、70°F和80°F。设计梯形图程序，满足以下需求：温度低于50°F时，3个加热器都打开；温度位于50°F～60°F范围内时，打开2个加热器；温度位于60°F～70°F范围时，打开1个加热器；温度高于80°F时，有一个确保3个加热器关闭的安全措施，以防产生故障时打开一个加热器。还需要一个主开关来控制系统的开关。为这个控制过程设计一个典型的PLC程序。
10. 用一个泵来填充2个存储槽。泵由START/STOP站的操作员手动启动。当第一个槽满时，控制逻辑必须能够通过使用传感器和电螺线阀自动停止向第一个槽填充，而改为填充第二个槽。当第二个槽满时，泵必须自动关闭。每个槽满时，相应的信号指示灯必须亮。
    a. 画出该过程的草图
    b. 写出该控制过程的典型PLC程序
11. 为下面的事件写出最合适的梯形逻辑过程，并写出使性能最优的指令。
    a. 如果限制开关LS1、LS2或LS3开，或者LS5和LS7开，则打开；否则关闭。（一般情况下，如果LS5和LS7开，其他情况就很少发生。）

    169

    b. 当开关SW6、SW7和SW8都开时，或者当SW55开时，打开输出。（SW55是警报指示，因此很少开。SW7大部分时间都开，其次是SW8，最后是SW6。）

第 7 章　定时器编程

学习目标：

- ❑ 描述气动通延时继电器和断延时继电器的运行方式。
- ❑ 描述 PLC 定时器指令且能区分非保持式定时器和保持式定时器。
- ❑ 将时间继电器原理图转换为 PLC 逻辑梯形图程序。
- ❑ 分析和解释典型 PLC 的定时器梯形图程序。
- ❑ 使用定时器指令的控制位编程实现对输出的控制。

除了线圈和触点，最常用的 PLC 指令为定时器。本章介绍了定时器怎样处理时间间隔和它们控制输出的方式，对 PLC 通延时继电器的基本功能及由它派生的其他定时功能做了讨论。文中还讨论了典型的工业定时任务。

7.1　机械式时间继电器

工业控制系统至少需要一种或两种定时功能。在电路的控制当中，用机械式时间继电器来实现对触点闭合和断开的延迟。除了机械式时间继电器的一些触点在线圈接通或断电之后的一定设定时间隔后才动作外，机械式时间继电器和控制继电器的运行方式很相似。

图 7-1 显示了一个气动通延时继电器的结构。时间延迟功能取决于空气通过固定孔的速度。通过对指针位置的定位来改变固定孔的个数，然后就可以调整时间延迟的长短。当线圈通电时，延时触点的闭合或断开动作就得到延迟。然而，当线圈失电时，延时触点瞬间返回到普通状态。特殊气动定时器除了具有延时触点外，还具有非延时触点。非延时触点和通用控制继电器的触点一样，直接受控于定时器的线圈。

机械式时间继电器通过两种方式来进行时间延迟。第一种方式为**通电延时**（如图7-1所示），即当继电器得电时开始时间的延迟。第二种方式为**断电延时**，即当继电器失电时开始时间延迟。图 7-2 展现了继电器延时触点的标准图解符号。

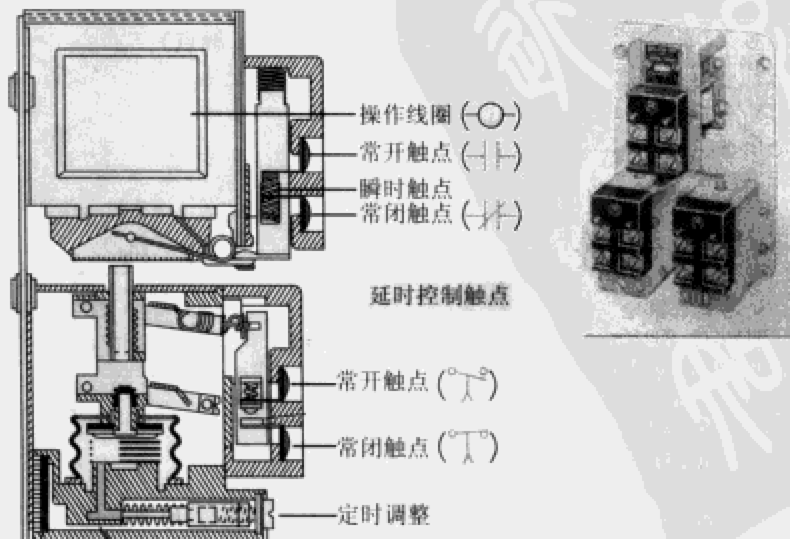


图 7-1　气动通延时继电器（得到 Allen-Bradley 公司许可）

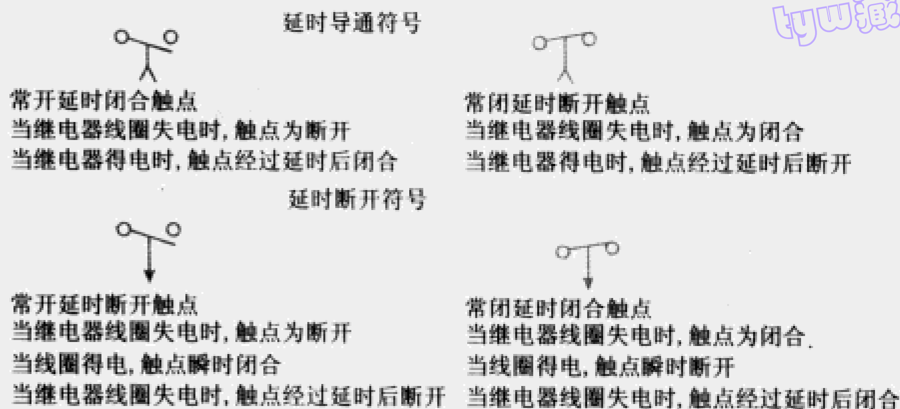
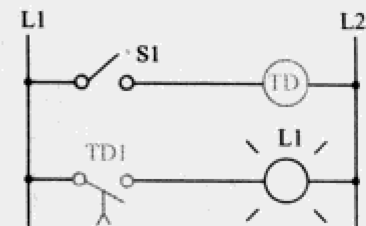


图 7-2　延时触点符号

171~172

为了说明基本的延时触点的功能，本书设计了图 7-3、图 7-4、图 7-5 和图 7-6 所示电路。在每个电路中，假定时间延迟的设定值为 10s。



操作顺序

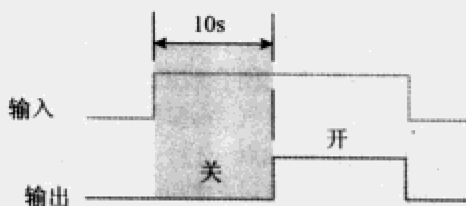
S1断开，TD失电，TD1断开，L1失电

S1闭合，TD得电，定时区间开始，TD1仍为断开，L1仍为失电

10s过后，TD1闭合，L1得电

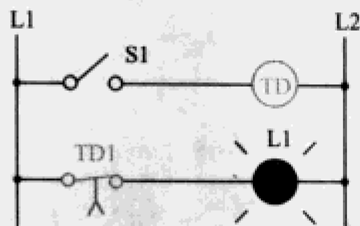
S1断开，TD失电，TD1立即断开，L1失电

(a) 操作



(b) 时序图

图 7-3　通延时继电器电路（NOTC 触点）



操作顺序

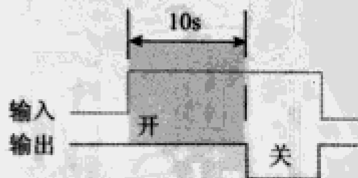
S1断开，TD失电，TD1闭合，L1得电

S1闭合，TD得电，定时区间开始，TD1仍为闭合，L1仍为得电

10s过后，TD1断开，L1失电

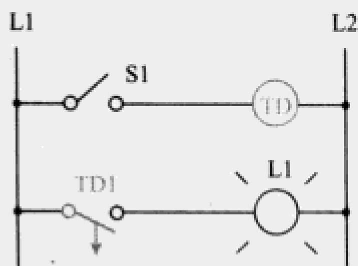
S1断开，TD失电，TD1立即闭合，L1得电

(a) 操作



(b) 时序图

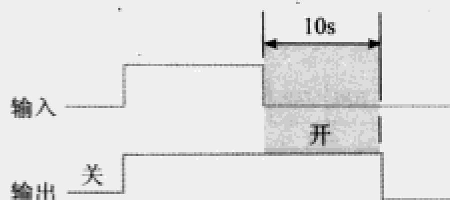
图 7-4　通延时继电器电路（NCTO 触点）



操作顺序

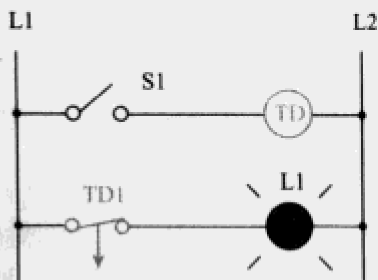
S1断开，TD失电，TD1断开，L1失电
S1闭合，TD得电，TD1立即闭合，L1得电
S1断开，TD失电，定时区间开始，TD1仍为闭合，L1仍为得电
10s过后，TD1断开，L1失电

(a) 操作



(b) 时序图

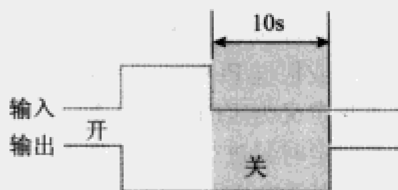
图 7-5　断延时继电器电路（NOTO 触点）



操作顺序

S1断开，TD失电，TD1闭合，L1得电
S1闭合，TD得电，TD1立即断开，L1失电
S1断开，TD失电，定时区间开始，TD1仍为断开，L1仍为失电
10s过后，TD1闭合，L1得电

(a) 操作



(b) 时序图

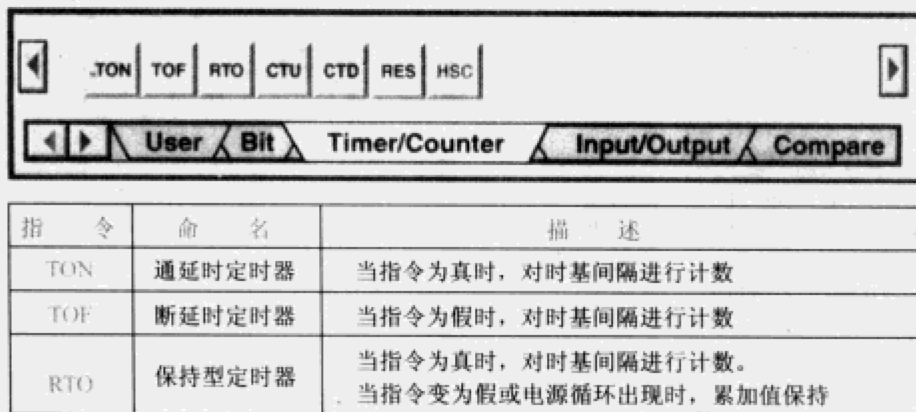
图 7-6　通延时继电器电路（NCTC 触点）

7.2　定时器指令

PLC 定时器作为输出指令，提供的功能同机械式时间继电器一样。它们都是用来在一定的时间间隔后控制设备运行或停止的。除了在第 6 章讨论的标准继电器指令之外，定时器和计数器指令也是 PLC 中的一对广泛使用的指令。虽然第一代 PLC 体系没有包含这些指令，但是现在生产的所有 PLC 中都具有这些指令。编程所用定时器的数量取决于所使用的 PLC 的型号，但一般都会远远超过需求数量。

PLC 定时器的优势在于能通过对程序的改动而方便地改变它的参数，或者能增加或减少它们在电路中的使用数量，而不用对接线进行改动。定时器的地址通常已被 PLC 的制造商指定下来，并且定位在一个指定的数据结构表区域。由于 PLC 定时器是基于固态技术，因此其另一个优势就是其精确度和可重复性非常高。

通常，有三种不同的定时器：通延时定时器（TON）、断延时定时器（TOF）及保持型定时器（RTO）。通延时定时器是基本指令，也是最常用的。通过一个或多个基本定时器指令可以衍生出大量的其他定时方式。图7-7展示了基于 Allen-Bradley SLC-500 PLC 及其相关 RSLogix 软件的典型程序定时器指令。



指令	命名	描述
TON	通延时定时器	当指令为真时，对时基间隔进行计数
TOF	断延时定时器	当指令为假时，对时基间隔进行计数
RTO	保持型定时器	当指令为真时，对时基间隔进行计数。 当指令变为假或电源循环出现时，累加值保持

图7-7　基于 Allen-Bradley SLC-500 PLC 及其相关 RSLogix 软件的定时器指令

与定时器相关的量：

- 预置时间表示定时电路中的持续时间。例如，如果需要延时10s，定时器将预置10s。
- 累计时间表示从线圈得电开始计时所得的实际消耗时间的量。
- 一旦定时梯级通电，定时器将开始以基本时间间隔计时，直到其预置值和累计值相等，或根据 PLC 的型号到达其最大定时时间，然后停止计时。定时器超时的间隔通常定义为定时器的时基。每个定时器都具有一个时基。可以利用几种不同的时基对定时器进行编程，如1s、0.1s 及 0.01s 等典型的时基。如果一个定时器程序的时基为0.1s 且延时累加次数为50，则定时器将延时5s（50×0.1s=5s）。

PLC 中的定时器通过处理模块中的内部生成时钟来运行。当 PLC 处于高频控制条件下或者需要 PLC 产生窄脉冲时，0.01s（10ms）的定时器是有用的。当 PLC 具有在非常长的用户程序和较多扫描次数时，10ms 的定时器可能就会带来一些问题。为了克服这些问题，在10ms 定时器或其他设备中，可以在程序中插入至少一个较长的扫描周期。附加梯级将会确保设备在累计时间之内被处理器扫描完全。

虽然在逻辑梯形图程序中各个制造商表示定时器的方法可能不一样，但多数定时器的运行方式都相同。可以将定时器指令描述为一个同机械时间继电器相似的继电器线圈（如图7-8所示）。定时器被分配一个地址，且等同于定时器。定时器的时基，定时器预置值或延时时间，定时器的累计值或当前延时时间同样为定时器指令的一部分。当定时梯级逻辑导通时，定时器开始根据时基与时基数运行，直到累计值等于预置值。当累计时间等于预置时间时，开始输出信号且与输出相关的触点闭合。通过例如 NO 或 NC 触点的程序，延时触点可以根据要求使用任意多次。

在逻辑梯形图当中，定时器通常由一个框来表示。图7-9说明了对于要求两根输入线的保持型定时器的方框格式。定时器模块具有控制和复位两个输入条件。控制线控制定时器实际的定时操作。当这根线上有信号或电源向输入端供电，定时器都将开始计时。把电源从控制端移去，将会中止定时器的进一步工作。

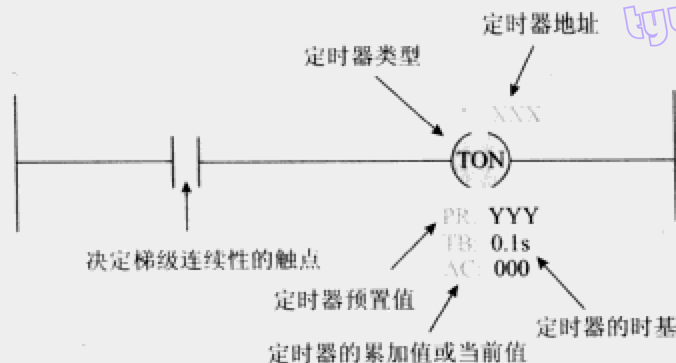


图 7-8　线圈形式的定时器指令

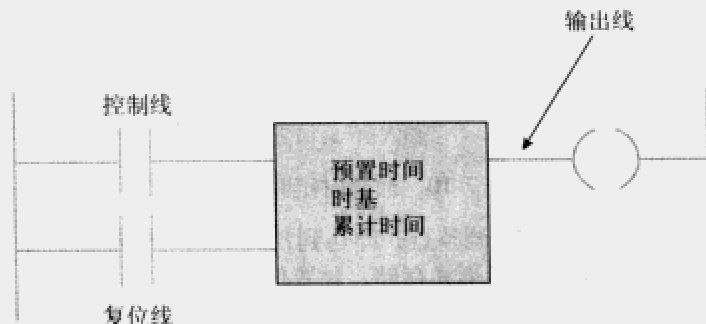


图 7-9　模块结构的定时器指令

复位线可以对定时器的累计值清零。一些制造商需要控制和清零两根线同时导通，定时器才开始工作；将电源从复位端移去，会使定时器清零。另外一些 PLC 制造商要求电能只需流入控制端而不需要流入复位端，就能使定时器开始运行。对这种型号定时器的运行，当复位端导通时，定时器将清零。

定时器指令模块包含属于定时器的运行信息，它包括预置时间、定时器的时基和当前时间及累计时间。所有模块格式的定时器都提供至少一种输出信号。定时器不停地将当前时间与预置时间进行比较，且当前时间小于预置时间时，定时器输出端为逻辑 0。当前时间等于预置时间时，输出端变为逻辑 1。

7.3　通延时定时器

定时器作为输出指令，其输入条件可以用检查是否断与检查是否通等指令。通过运行逻辑程序可以确定定时间隔。当含有定时器的梯级得电，通延时定时器开始运行，开始进入定时区间。在定时器定时区间末，输出变为 1，如图 7-10 所示。在定时器梯级得电的一段时间之后，定时器输出才变为 1；因此，我们说定时器实现了延时。时间延时的长度可以通过改变预设值来进行调节。此外，大多数 PLC 都允许改变定时器的时基或分辨率。时基越小，定时器精度越高。

Allen-Bradley PLC-5 和 SLC-500 控制器中的每个定时器单元都占据三个数据表字：控制字、预置字和累加字。控制字有三个控制位：

❑ **使能（EN）位**

当定时器指令为真时，则使能位为真（状态为 1）。当定时器指令为假时，使能位为假（状态为 0）。

❑ **正在定时（TT）位**

当定时器的累加值发生改变时，意味着定时器正在计时，则正在定时位为真。当定时器没有计时时，累加值就不会发生改变，因此正在定时位为假。

❑ **完成（DN）位**

当累加值达到预置值时，完成位的状态就发生改变。其状态取决于所使用的定时器的类型。

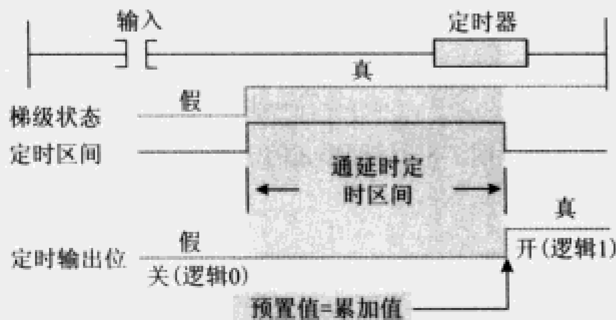


图7-10　通延时定时顺序

预置值（PRE）字为定时器的调整点，即达到预置值则定时器就开始定时。预置值字的范围为0到32 767且以二进制的形式存储。预置值中不会出现负数。

累加值（ACC）字为一个随着定时器计时而不断增加的值。当累加值达到预置值时，它就停止增长。

定时器指令同样需要人为地输入一个时基，它可以是1.0s或0.01s。实际预置时间间隔是将时基与存储在预置值字中的值相乘所得到的。实际累加时间间隔是将时基与存储在累加值字中的值相乘所得到的。

图7-11显示了一个通延时定时器指令作为Allen-Bradley PLC-5和SLC-500控制器指令集的一部分运用的例子。输入的信息包括：

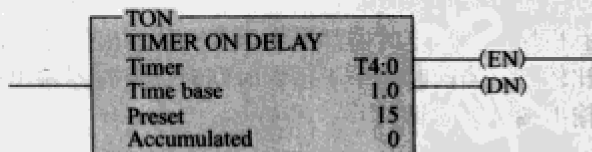


图7-11　通延时定时器指令

❑ **定时器编号**

定时器编号必须来自定时器文件。如例中所示，定时器编号为T4:0，表示0号定时器在4号定时器文件夹中。每个定时器文件夹中可以有多达1000个定时器，编号从0到999。每个定时器必须有独立的地址且地址不适用于其他定时器。

❑ **时基**

时基（通常以秒来表示）可以为1.0s或0.01s。如例中所示，时基为1.0s。

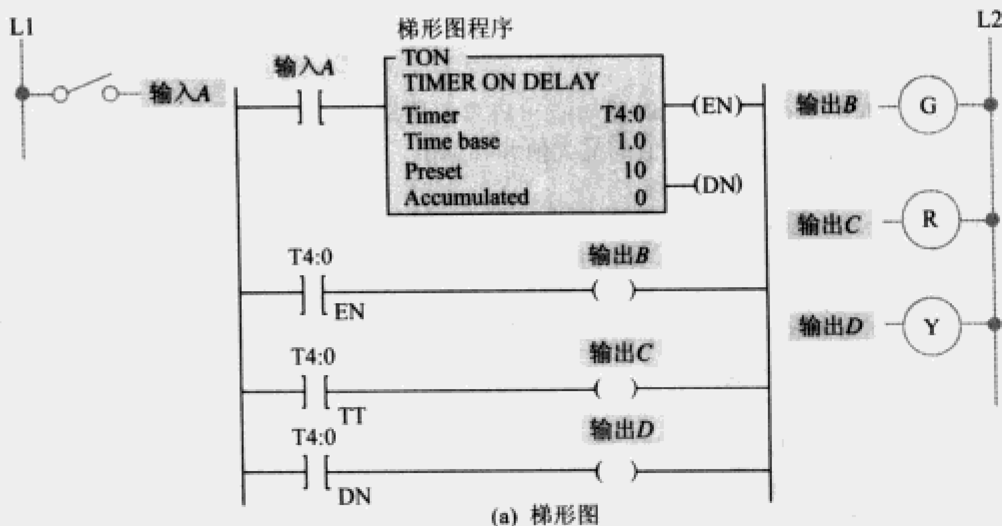
❑ **预置值**

如例中所示，预置值为15。预置值的变化范围可以从0到32 767。

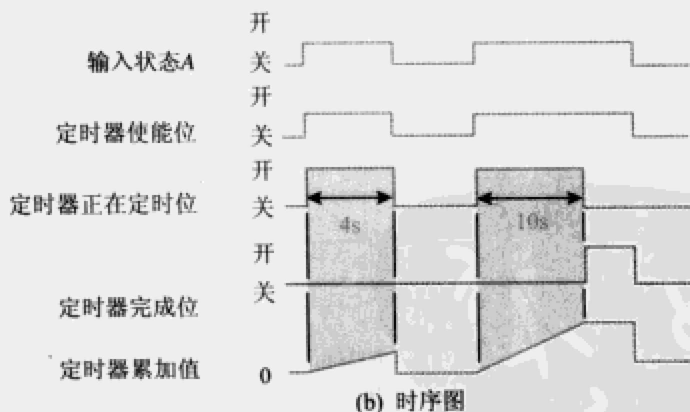
❑ **累加值**

如例中所示，累加值为0。虽然有可能输入一个0到32 767之间的任一个数，但定时器的累加值通常按0输入。不考虑以前载入的值，当定时器被置位时，定时器的值变为0。

通延时定时器（TON）是最常用的定时器。图 7-12 展现了一个使用了通延时定时器的 PLC 程序。当开关闭合时，定时器被激活。定时器预置时间为 10s，当到达 10s 后，输出端 D 将接通。当开关闭合时，定时器开始计时，直到累加值等于预置值时停止；然后输出端接通。如果在定时器还没有计时完的时候开关就断开，则累加时间自动清零。由于掉电而引起定时器指令置位，我们把定时器的这种结构称为非保持型。这种定时器的运行方式为通延时定时，这是由于开关从断开状态变为闭合状态之后，又经过了 10s 输出端 D 才闭合。



(a) 梯形图



(b) 时序图

定时器元素

15	14	13	12 11 10 9 8 7 6 5 4 3 2 1 0	字
EN	TT	DN	Internal use	0
Preset value PRE				1
Accumulated value ACC				2

可寻址位

EN=Bit 15 enable
TT=Bit 14 timer timing
DN=Bit 13 done

可寻址字

PRE=Preset value
ACC=Accumulated value

(c) 含三个字的元素。字0为控制字，字1存储预置值，字2存储累加值(Allen-Bradley PLC-5和SLC-500格式)

图 7-12 通延时定时器

在图7-12b中，时序图首先展现了定时器定时4s后失电的情况。定时器置位且正在定时位和使能位都同时为假，累加值同样被清零。然后输入端A重新得电且保持真状态超过10s。当累加值达到10s后，执行位（DN）从假变为真，且正在定时位（TT）从真变为假。当输入端A变为假时，定时器指令也变为假并且置位，同时控制位全部清零且累加值清零。

178

在Allen-Bradley PLC-5和SLC-500中，定时器地址通过以下三种不同的层次进行操作：元素层、字层和位层。在每个元素中，定时器用了三个字。如图7-12c所示，每个元素包含一个控制字、一个预置字和一个累加字。每个字作为一个子单元都是一个元素地址。每个整数元素具有一个字。每个字长16位，从0到15。当对位层进行编址时，地址通常涉及位里面的字或子元素。

Allen-Bradley Controllogix定时器功能和PLC-5及SLC-500控制器的功能一样。不同之处在于时基、定时器地址、预置值的最大值和累加值。对于逻辑控制器：

- 时基固定为1ms（0.001s）。
- 地址为一个定时器数据型的预定义结构。
- 预置值的最大值和累加值分别为2 147、483 647。

定时器可能产生一个和它们自身相关的瞬时输出（通常也叫使能位）信号。如果需要定时器产生一个瞬时输出信号且定时器指令中并不提供这个信号，可以在程序中使用一个内部参考继电器线圈来等效瞬时触点指令。

图7-13显示了这项技术的运用。根据继电器梯形原理图，当按下开始按钮5s之后，线圈M得电。触点1TD-1为瞬时触点，触点1TD-2为延时触点。梯形图程序展现了用一个触点指令来等效一个内部继电器，通过此技术实现定时器运行。瞬时触点等效为内部继电器线圈，而延时触点被等效为延时输出线圈。

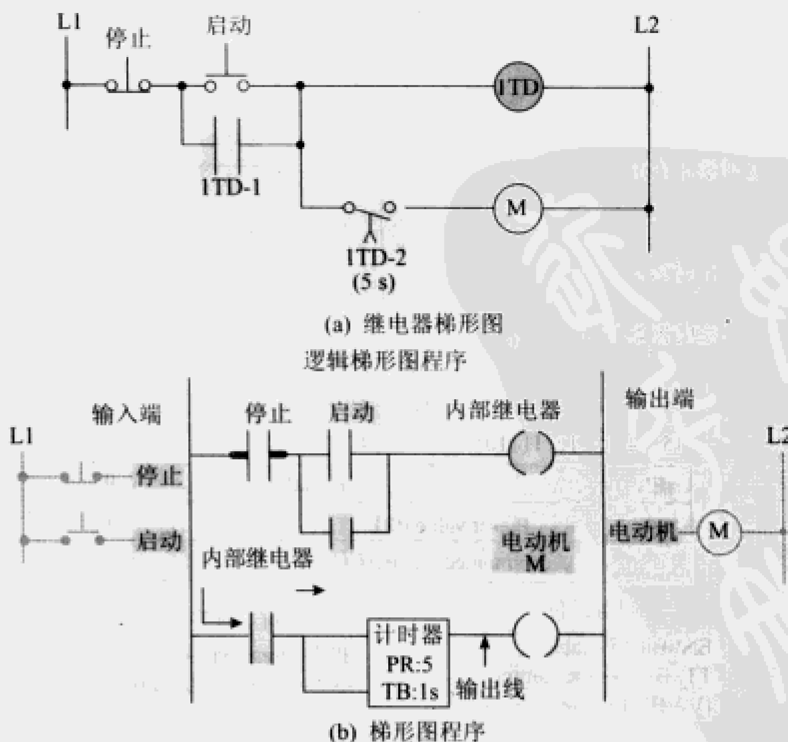
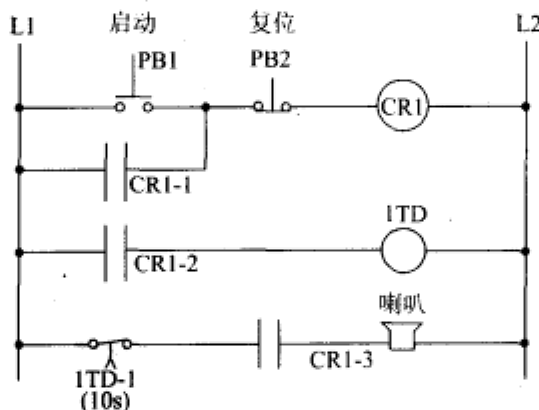
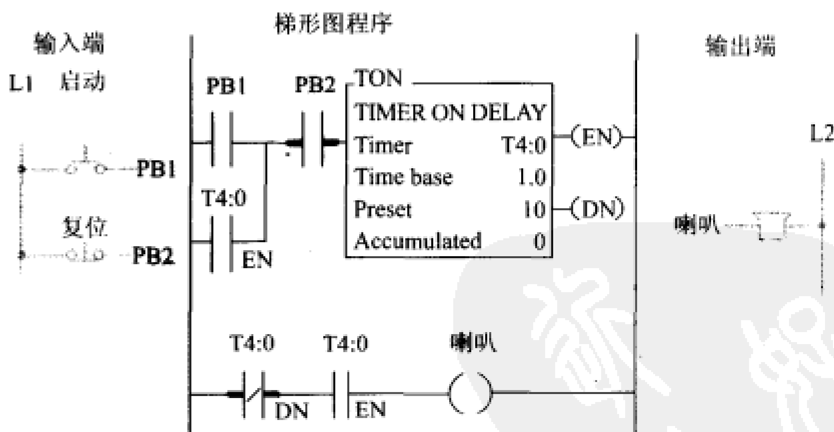


图7-13　利用瞬时输出程序实现通延时定时器

图 7-14 显现了使用 NCTO 触点的通延时定时器的应用。此电路的作用是作为移动设备的报警信号，例如关于传送器电机的启动。根据继电器梯形原理图，当启动按钮 PB1 被按下时，线圈 CR1 就被接通。因此，触点 CR1-1 闭合使得线圈 CR1 自保持，触点 CR1-2 闭合使得定时器线圈被接通，触点 CR1-3 闭合使得喇叭发出声音。在经过了 10s 的延时之后，定时器触点 1TD-1 断开，自动关闭喇叭。梯形图程序显示了怎样利用 PLC 来编写程序。 [179]



(a) 继电器梯形图



(b) 梯形图程序

图 7-14 启动报警信号电路

图 7-15 显示了使用逻辑控制 TON 指令的通延时定时器的应用。此程序要求当开关闭合后延时 12s，螺线管操作阀得电。 [180]

定时器通常作为自动顺序控制系统的一部分。图 7-16 显示了怎样通过一个启动/停止控制台实现一系列电机的自动启动。根据继电器梯形原理图，当启动按钮 PB2 在任一时刻瞬时导通时，润滑油泵电动机的启动线圈 M1 得电。因此，M1-1 控制触点闭合使得 M1 自锁，润滑油泵电动机启动。当润滑油泵建立起足够大的油压时，润滑油压力开关 PS1 闭合。然后线圈 M2 得电，使得主驱动电机启动并且线圈 1TD 得电，开始进入定时区间。在经过预置延时时间 15s 之后，1TD-1 触点闭合使得线圈 M3 得电且启动馈送电机。梯形图程序显示了怎样利用 PLC 对此电路进行编程。 [181]

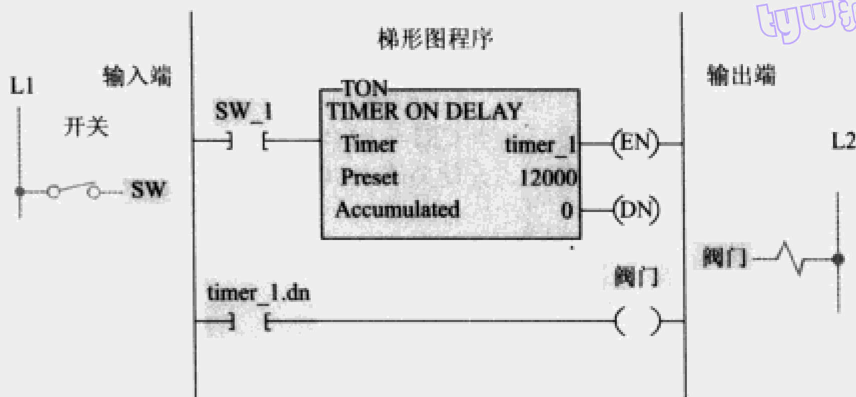
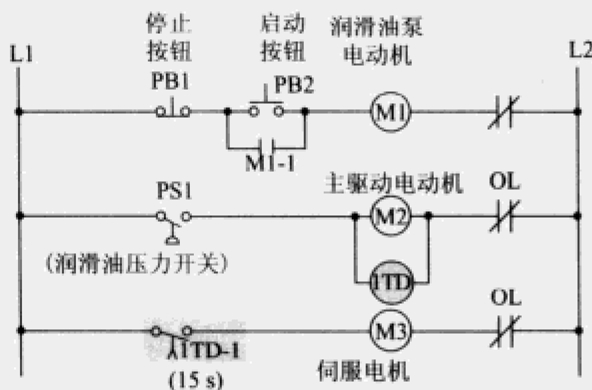
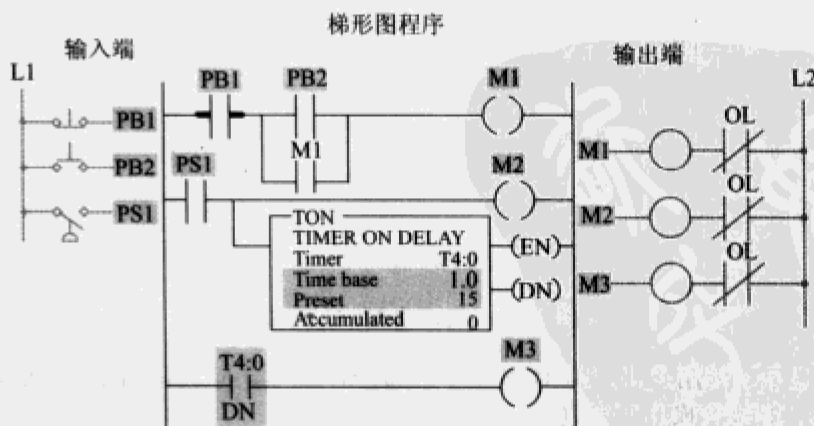


图 7-15　使用逻辑控制 TON 定时器编程实现螺线管操作阀的通电延时



(a) 继电器梯形原理图



(b) 梯形图程序

图 7-16　自动顺序控制系统

7.4　断延时定时器指令

当梯级中的定时器失电后，断延时定时器（TOF）将会在一段时间内继续保持输出信号。图 7-17 显示了使用 SLC-500 TOF 定时器指令的断延时定时器的通用程序。如果逻辑连续信号消失，定时器就开始对时基间隔进行计数，直到累加时间等于程序的预置时间。

当与输入端相连的开关 I:1.0/0 首先闭合，定时器输出端 O:2.0/1 立即置为 1 且指示灯发光。如果现在断开此开关，逻辑连续信号消失且定时器开始计时。15s 之后，当累加时间等于预置时间时，输出端置 0 且指示灯熄灭。如果在定时时间之内重新获得逻辑连续信号，累加时间就置 0。因此，这种定时器也属于非保持型定时器。 182

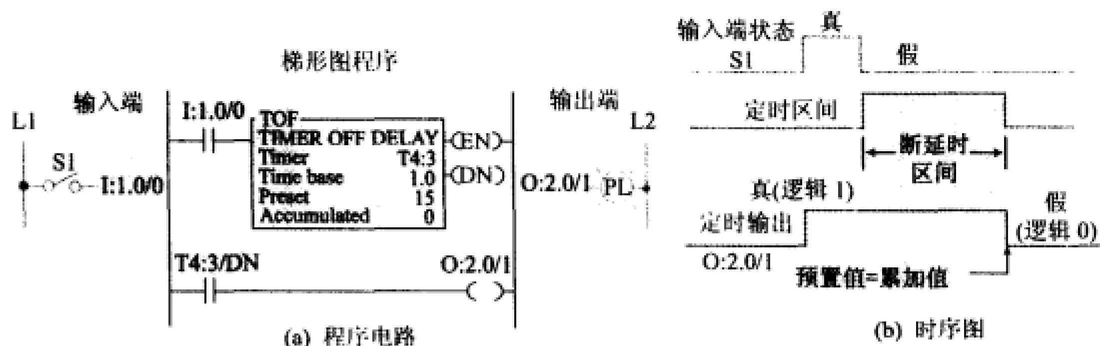


图 7-17 程控断延时定时器

图 7-18 显示了使用 ControlLogix 断延时定时器的指令。在其应用中，闭合开关将立即启动电机 M1、M2 和 M3。当开关断开时，电机将在 5s 后停下来。

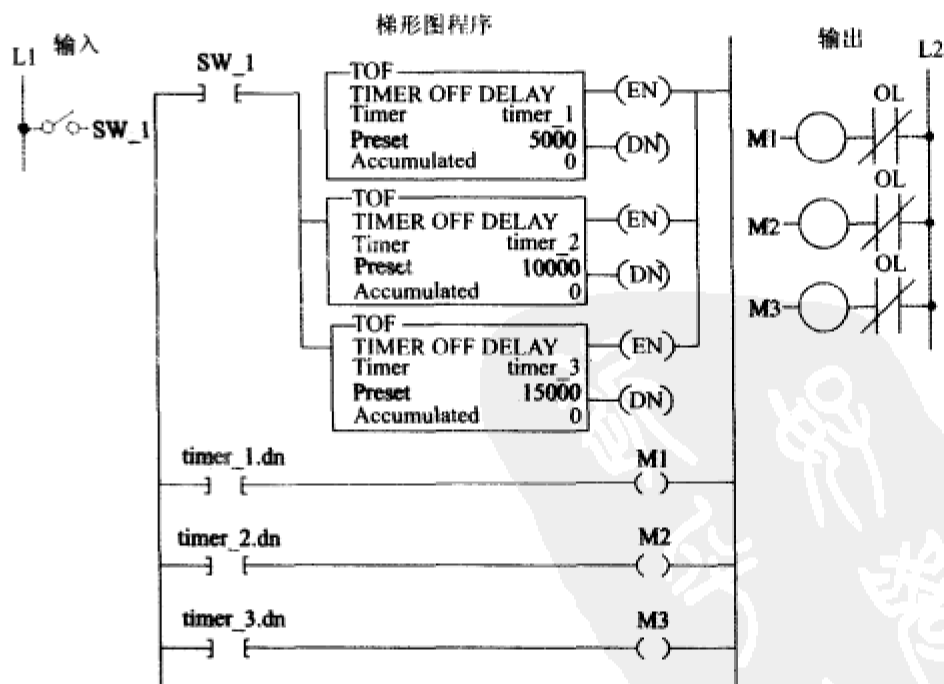
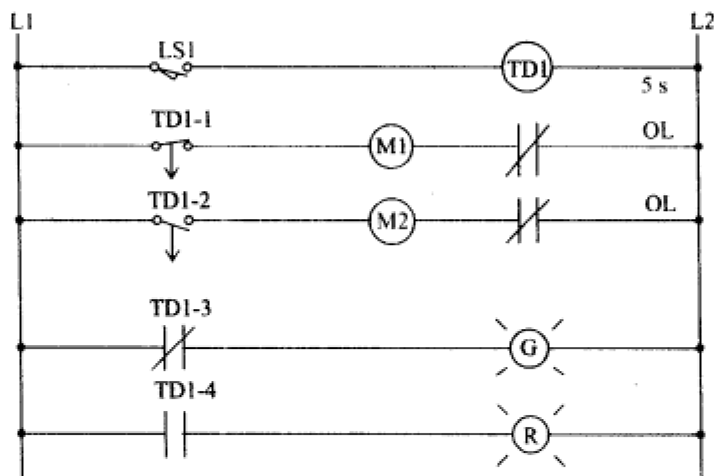


图 7-18 断延时定时器控制电机 5s 时延停机

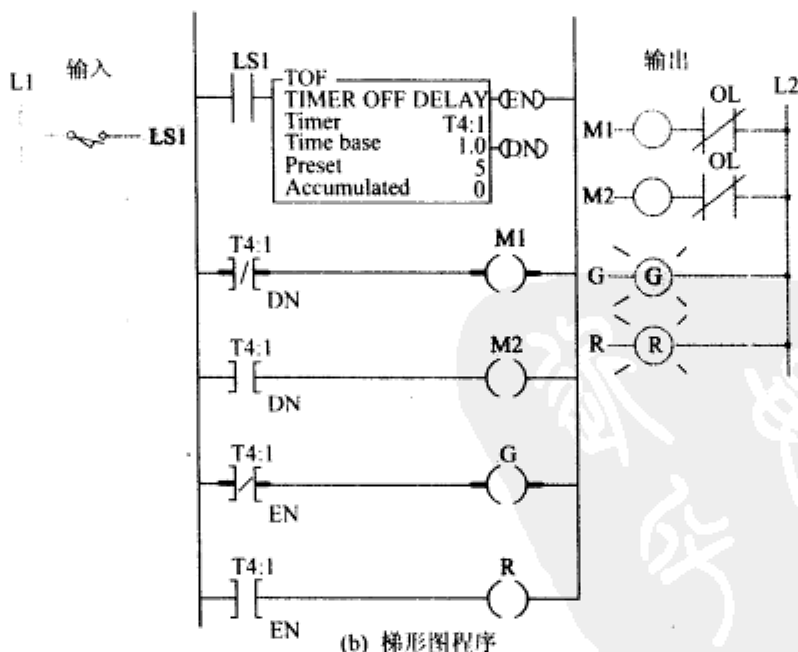
图 7-19 显示了怎样利用 PLC 对气动断延时定时器的继电器线路进行编程。根据继电器原理图，当电源首先被加上（限位开关 LS1 断开）时，电机启动线圈 M1 得电且绿色指示灯亮，同时线圈启动线圈 M2 失电且红色指示灯熄灭。

当限位开关 LS1 闭合时，断延时定时器线圈 TD1 得电。因此，延时触点 TD1-1 断开使得电机启动线圈 M1 失电，延时触点 TD1-2 闭合使得电机启动线圈 M2 得电，瞬时触点 183

TD1-3 断开使得绿色指示灯熄灭且瞬时触点 TD1-4 闭合使得红色指示灯亮。只要限位开关是闭合的，那么电路将一直保持这种状态。



(a) 继电器原理图



(b) 梯形图程序

图 7-19　气动断延时定时器的编程

当限位开关 LS1 断开时，断延时定时器线圈 TD1 失电。因此，延时区间开始，瞬时触点 TD1-3 闭合使得绿色指示灯亮，并且瞬时触点 TD1-4 断开使得红色指示灯熄灭。在 5s 延时区间过后，延时触点 TD1-1 闭合使得电机启动线圈 M1 得电，并且延时触点 TD1-2 断开使得电机启动线圈 M2 失电。图 7-19b 显示了怎样利用 SLC-500 TOF 定时器实现对电路的编程。

图 7-20 显示了一段同时使用了通延时定时器和断延时定时器的程序。整个过程为将油箱 A 中的液体抽到油箱 B 中。运行过程可描述如下：

- 启动之前，PS1 必须闭合。
- 当按下启动按钮后，油泵开始工作。当释放按钮后，油泵将继续运行。
- 当按下停止按钮后，油泵停止运行。
- 在油泵启动 5s 后，PS2 和 PS3 必须闭合。如果 PS2 或 PS3 断开，那么油泵将停止运行且需要经过 14s 之后才能再次启动。

(a) 过程

(b) 梯形图程序

图 7-20 液体抽取过程

7.5 保持型定时器

当设备得电时，保持型定时器将开始对时间进行累加，并且当定时器失电时，会保持当前计时值不变。一旦设备的累加时间等于它的预置时间，设备的触点将改变状态。在达到预置时间之后，设备失电将不会影响其触点状态。为了使累加时间置 0 和触点状态回到其原来状态，保持型定时器必须人为地施加一个离散信号来复位。

图 7-21 显示了电机驱动、电子机械式保持型定时器的在一些应用中的动作。轴固定

的凸轮由一个电动机驱动。当接上电源，电动机启动带动轴和凸轮旋转。凸轮突起物的位置以及电动机的减速装置，可以使电动机驱动凸轮时，有足够的时间来激发触点。如果移去电动机的电源，轴将停止旋转但不会置位。

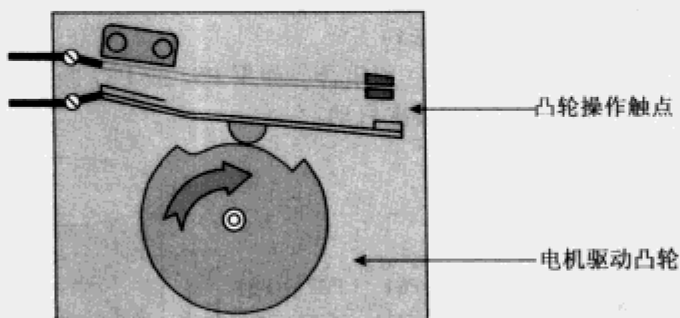


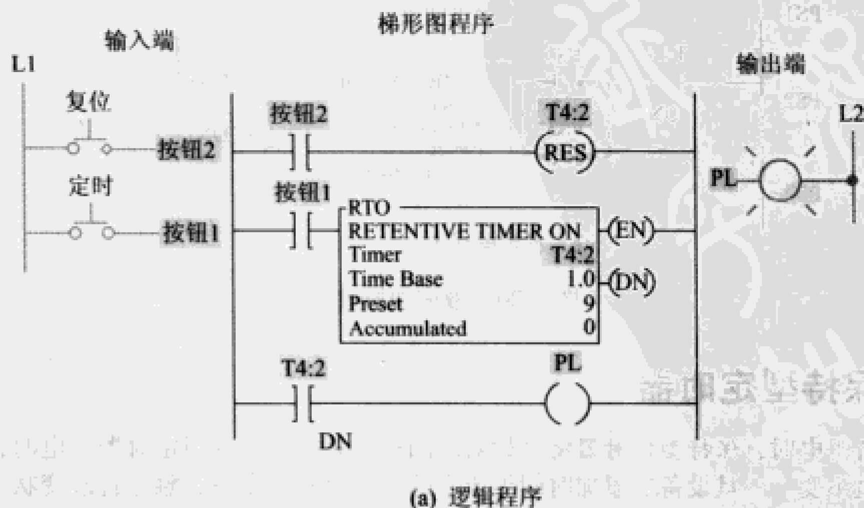
图 7-21　机电保持型定时器

185~186

PLC 保持型通延时定时器（RTO）的运行方式同非保持型通延时定时器（TON）相似，其主要的不同在于保持型定时器复位指令。不同于 TON 的是，当定时器梯级失电时，RTO 将保持它的累加值，并且当定时器梯级从新得电之后，RTO 将从计时中断处继续开始计时。这种定时器必须补充一个定时器复位指令使累加值置 0。RES 指令是唯一的一种使保持型定时器累加值复位的自动方式。RES 指令与它所复位的定时器指令具有相同的地址。当 RES 指令为真时，定时器累加值和定时器完成位（DN）都被置 0。

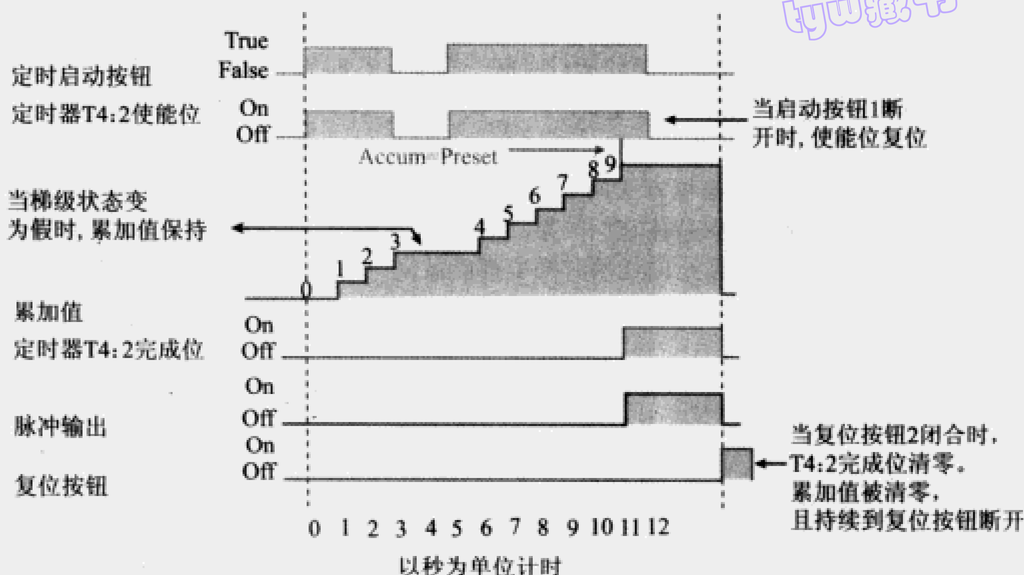
图 7-22 显示了伴有时序图的保持型定时器电路的 PLC 程序。当定时按钮 PB1 闭合时，定时器开始计时。如果 3s 后按钮断开，定时器的累加值将停止在 3s。当定时按钮重新闭合时，定时器将继续从 3s 开始计时。当累加值等于预置值，定时器的完成位 T4:2/DN 被置 1 且指示灯输出端 PL 接通。

当定时器失电时，由于保持型定时器没有置 0，因此需要使用复位指令 RES 来使定时器复位。RES 指令给出了同 RTO 一样的地址（T4:2）。当复位按钮 PB2 闭合，RES 使累加时间置 0 且 DN 位置 0，指示灯 PL 熄灭。



(a) 逻辑程序

图 7-22　保持型通延时定时器程序和时间图



(b) 时序图

图 7-22 （续）

图 7-23 所示的程序显示了 RTO 的实际应用。RTO 定时器的目的是为了检测一个管道系统处于过压状态的累计时间是否超过 60s。当达到 60s，喇叭将会自动报警以提醒人员对故障的注意。当维护人员注意到报警信号，他们可以使用钥匙开关将 S1 置于复位状态（触点闭合）使得报警信号消失。在故障解除后，可以通过钥匙开关将置于导通状态（触点断开），报警系统就能够再次启动。图示定时器代表一个逻辑控制画面，PLC-5 和 SLC-500 也以同样的方式工作。



图 7-23 保持型延时报警程序

图 7-24 显示了通延时、断延时和保持型定时器指令在同一个程序中的实际应用。在这个工业的应用当中使用了带有一根巨大钢轴支撑的机器，机器由巴氏合金轴支撑。钢轴和一个马达耦合。巴氏合金轴需要润滑，润滑油由一个小型马达驱动的油泵提供。运行步骤如下：

- 要启动机器，操作员将 SW 接通。
- 在马达的轴开始旋转之前，要通过系向合金轴提供 10s 的润滑油。
- 当机器开始运行，合金轴仍然接收润滑油。

[187]

- 当操作员断开 SW 使机器停机后，润滑油泵继续提供润滑油，此过程将持续 15s。
- 保持型定时器用于追踪油泵的运行时间。当总运行时间为 3h 时，关闭马达且指示灯亮以显示过滤器和油需要更换。
- 在完成了更换过滤器和油之后，通过复位按钮使过程复位。

保持型断延时定时器的编程方法同 RTO 相似。即使在定时器定时完成之前逻辑连续信号丢失或失电，二者都能保持住它们的累加时间值。保持型定时器不需要在时间耗尽的情况下才允许复位，而是在运行的任何时间里都能被复位。注意，即使定时器的控制输入端有逻辑连续信号，复位输入信号仍然能覆盖其控制信号。

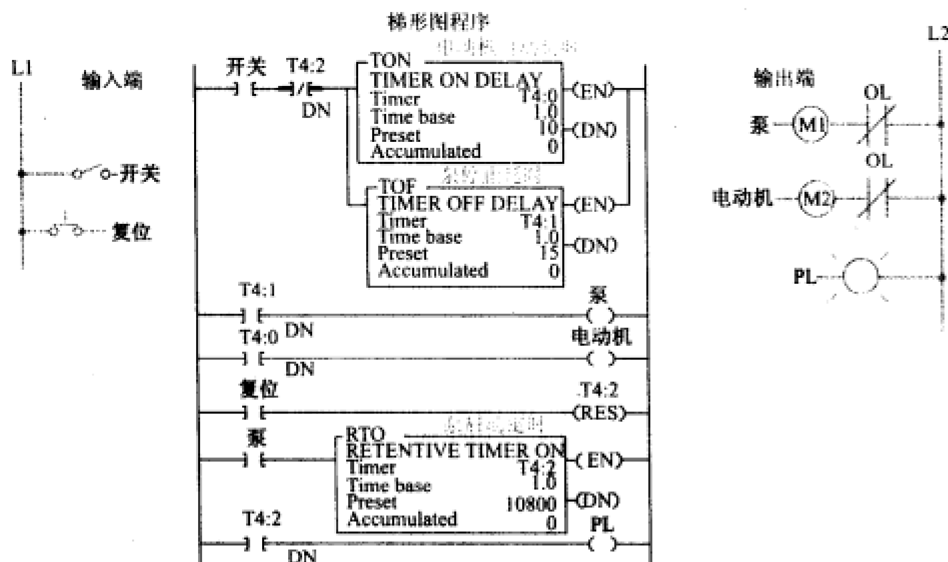
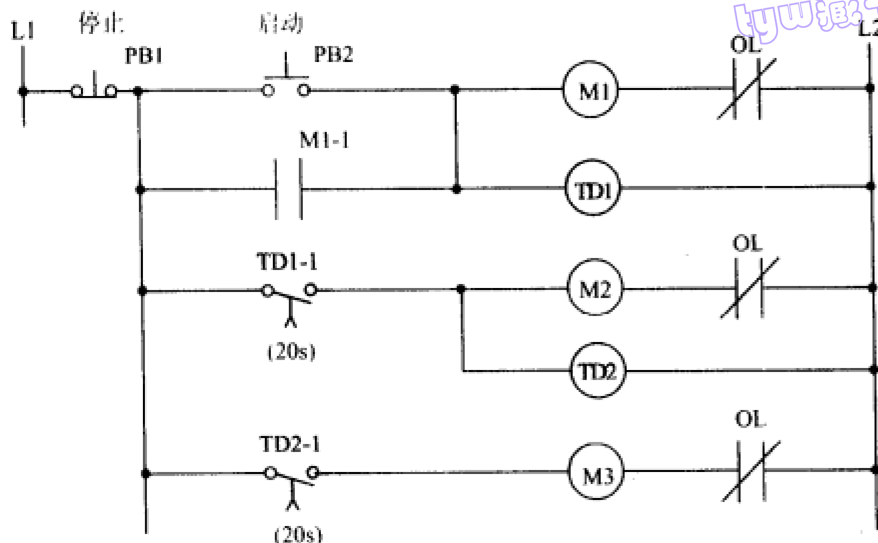


图 7-24　轴润滑程序

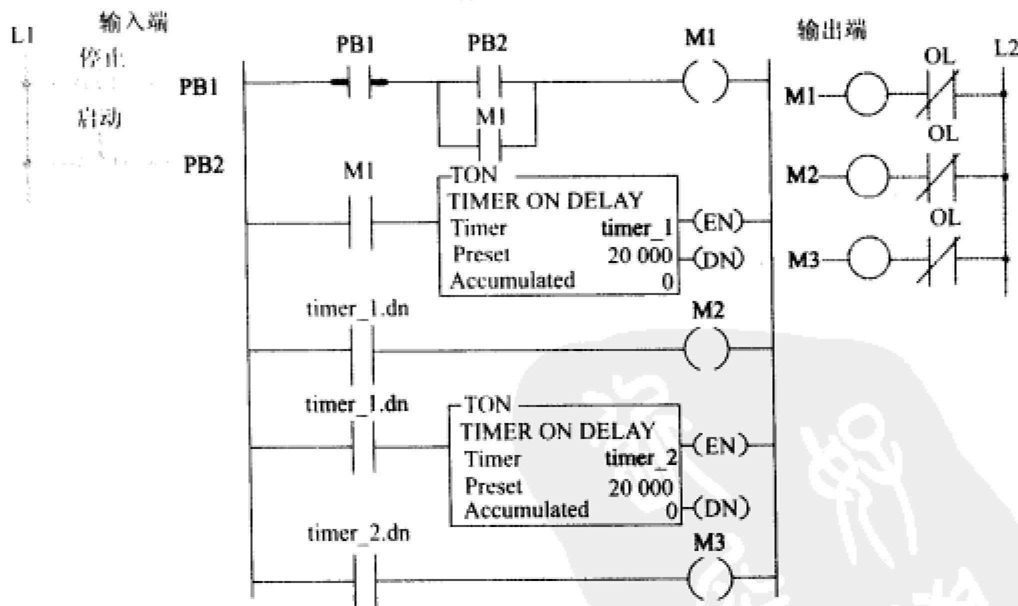
7.6　定时器的级联

在编程中将两个或更多的定时器合在一起称为级联。为了满足各种逻辑控制的需要，[188] 定时器之间可以互联或级联。图 7-25 显示了怎样使三个电动机顺序自动启动，相邻电动机启动之间相差 20s。根据继电器原理图，当启动按钮 PB2 瞬时导通，电动机启动线圈 M1 得电。因此，电动机 1 启动，触点 M1-1 闭合使得 M1 自锁，延时线圈 TD1 得电使得第一段定时区间开始。在 20s 预置时间之后，TD1-1 触点闭合使得电动机启动线圈 M2 得 [189] 电。因此，电动机 2 启动，延时线圈 TD2 得电，第二段定时区间开始。在经过 20s 预置时间后，TD2-1 触点闭合使得电动机启动线圈 M3 得电，因此电动机 3 启动。梯形图程序显示了怎样利用 PLC 进行编程。注意到使用了两个逻辑控制定时器，第一个定时器的输出信号用来控制第二个定时器的逻辑输入。

将两个定时器互联形成一个振荡器电路。逻辑振荡器基本上就是一个通过编程来产生任何宽度的周期性脉冲的定时电路。图 7-26 显示了信号器闪光电路的程序。两个内部定时器形成振荡电路，它能产生定时脉冲。通过编程，振荡器电路的输出与报警条件串联。如果报警条件（温度、压力或限位开关）为真，输出信号指示灯将开始闪烁。注意可以使用同样的闪光电路实现任何数量报警条件的编程。



(a) 时延概图



(b) 逻辑梯形图

图 7-25　电动机的延时顺序启动电路

在使用 PLC 时，有时我们需要一个延时区间比最大允许预置时间更长的单独定时器指令。在这种情况下，可以通过如图 7-27 所示的定时器级联来解决问题。在这个例子中，编程所用定时器的类型为 TON，总定时时间要求 42 000s。第一个定时器 T4:1 的预置时间设为 30 000s，且当输入端 SW 闭合时开始计时。当定时区间 30 000s 完成，T4:1/DN 位将置 1。这将会启动第二个定时器 T4:2，其预置时间设为 12 000s。一旦 T4:2/DN 达到预置时间，T4:2/DN 位将置 1，接通指示灯输出信号 PL，显示 42 000s 的定时时间已经全部完成。在任何时间断开输入端 SW 将使两个定时器都复位，且输出信号关闭。 [190]

一个典型的PLC定时器应用就是交通灯控制。图7-28中的梯形图电路显示了在只有一个方向情况下一组交通灯的模拟控制。通过定时器级联电路实现红灯到绿灯到黄灯的转换。运行过程如下：

红	30s
绿	25s
黄	5s

191

此过程将不断重复。图7-29显示了将原来的交通灯改进为又增加了三个灯的系统，使之能控制另一个方向的交通。

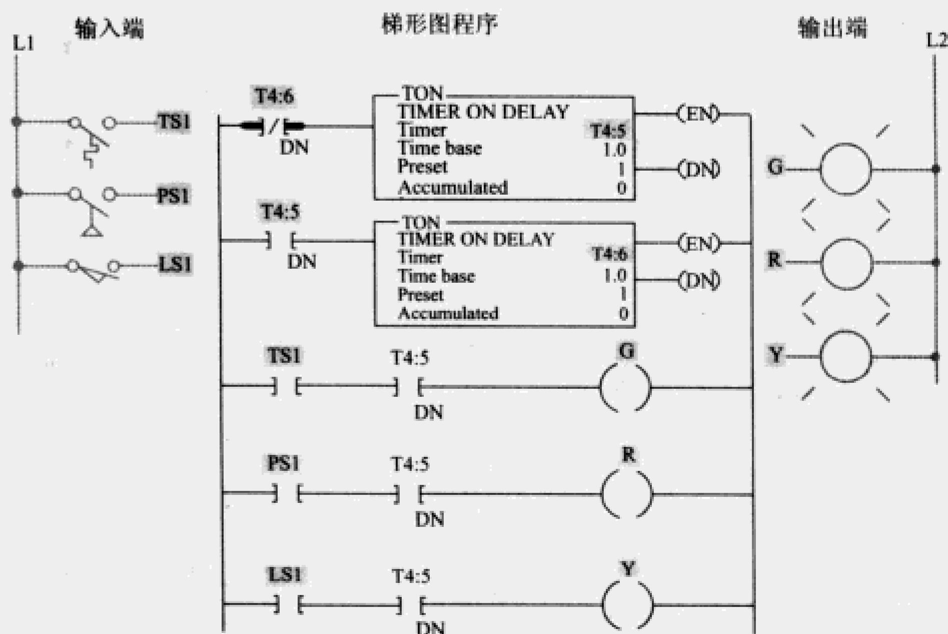


图7-26　信号器闪光程序

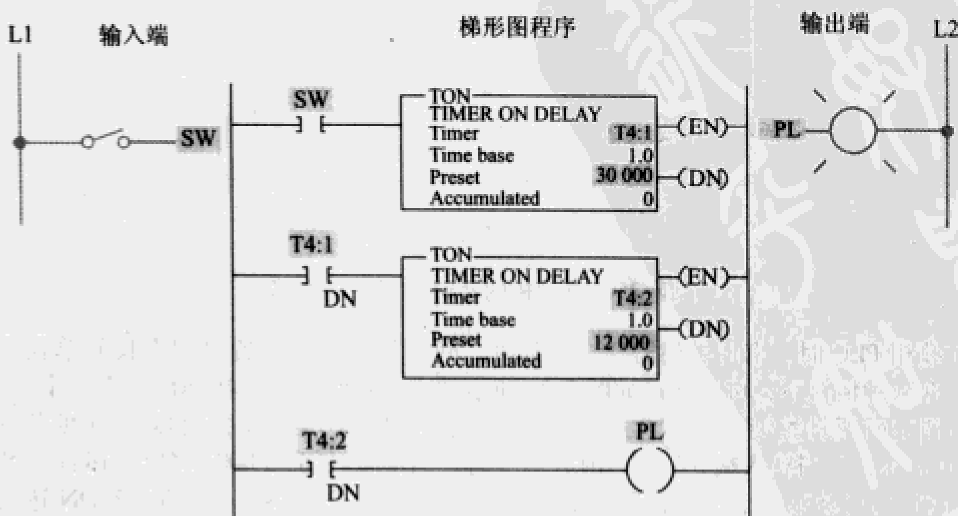


图7-27　通过定时器的级联实现更长的延时

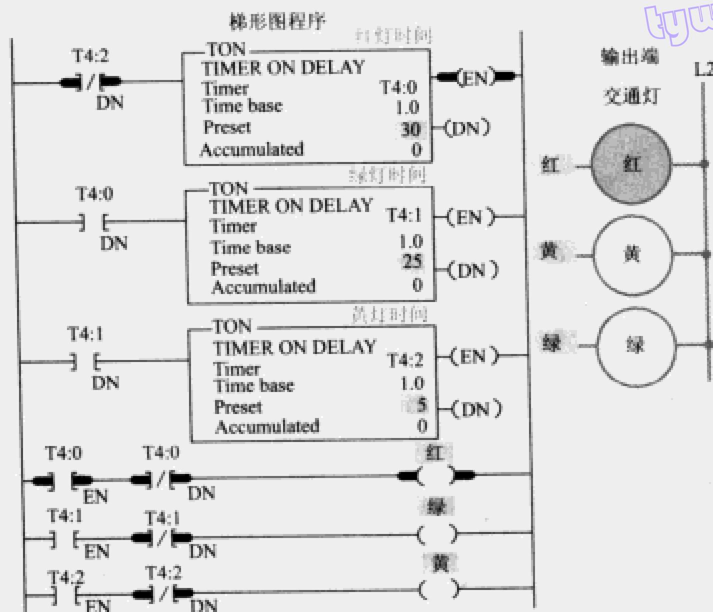


图 7-28 单方向的交通灯控制

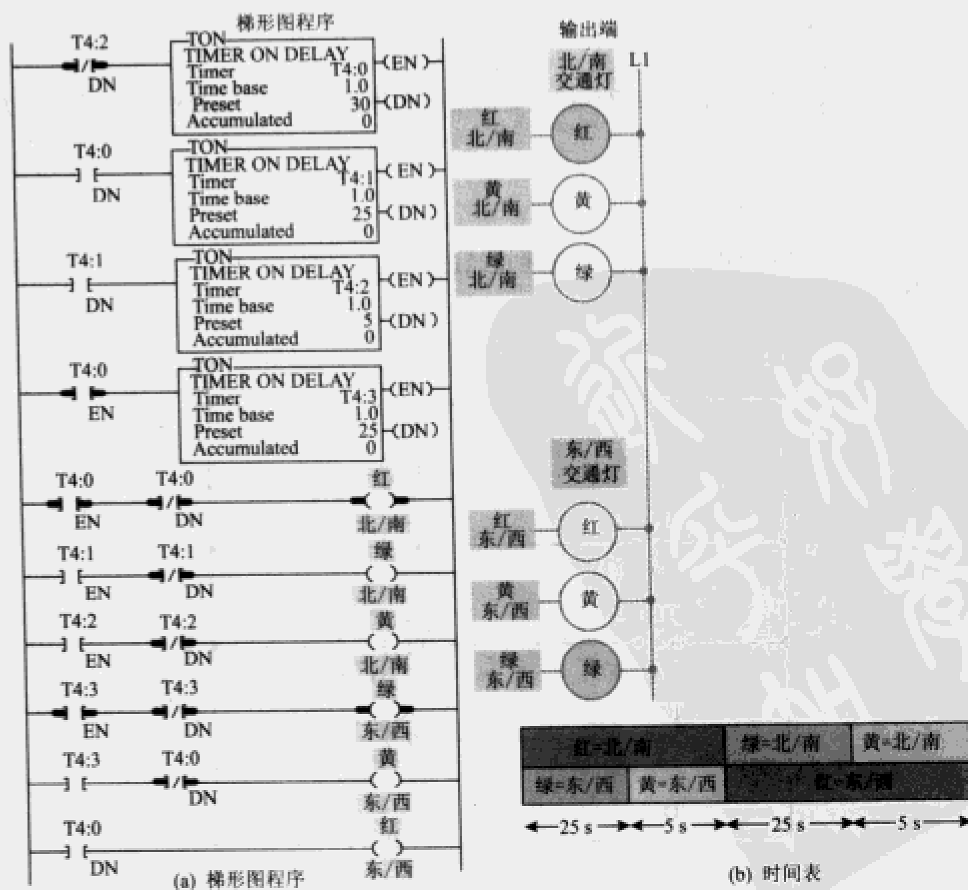


图 7-29 双向交通灯控制

思考题

1. 解释气动定时器的延时触点和瞬时触点之间的差别。
2. 画出以下气动定时器的触点符号并解释其工作原理：
   a. 通延时定时器——NOTC触点　　b. 通延时定时器——NCTO触点
   c. 断延时定时器——NOTO触点　　d. 断延时定时器——NCTC触点
3. 陈述五条关于常用的与PLC定时器指令相关的信息。
4. 什么时候程控定时器的输出端得电？
5. 在PLC梯形图程序中表示定时器的两个常用方法是什么？
6. a. 解释保持型定时器和非保持型定时器之间的区别。
   b. 解释怎样将保持型定时器和非保持型定时器的程序累加计数置0。
7. 陈述三个使用程序PLC定时器的优点。
8. a. 写出三种不同类型PLC定时器的名字。
   b. 这三种不同类型的定时器中哪一种最常用？
9. 解释以下关于PLC定时器指令的参数各代表什么：
   a. 预置时间　　b. 累加时间　　c. 时基
10. a. 什么时候定时器指令的使能位为真？
    b. 什么时候定时器指令的正在定时位为真？
    c. 什么时候定时器指令的完成位为真？
11. 解释以下各种定时器的累加时间复位的方法：
    a. TON定时器　　b. TOF定时器　　c. RTO定时器
12. 对Allen-Bradley PLC-5和SLC-500控制器的定时器编址方法和ControlLogix的编址方法进行比较。

194

习题

1. a. 根据图7-30中的继电器原理图，陈述在经过以下各顺序事件后的指示灯状态（打开或闭合）：

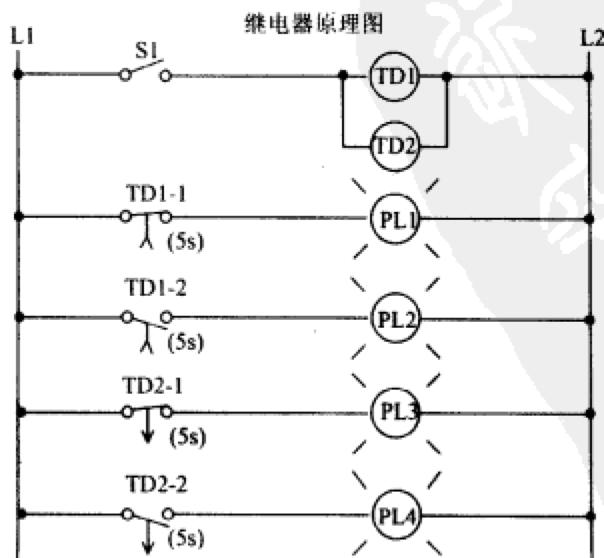


图　7-30

(1) 接通电源，开关 S1 打开。
(2) 开关 S1 闭合。
(3) 开关 S1 闭合后经过 5s。
(4) 开关 S1 打开。
(5) 开关 S1 打开后经过 5s。

b. 设计一个 PLC 程序，画出标准 I/O 连接图和梯形图程序，使硬接线控制电路能正确运行。

2. 设计一个 PLC 程序，画出标准 I/O 连接图和逻辑梯形图程序，使如图 7-31 所示的硬接线控制电路能正确运行。

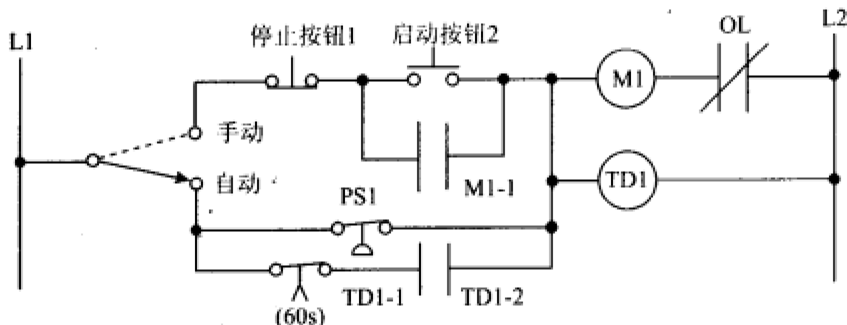


图　7-31

3. 研究图 7-32 所示的梯形图程序，回答以下问题：

a. 程序使用了什么类型的定时器？
b. 延时区间的长度为多少？
c. 当第一次通电时，定时器累加值为多少？
d. 什么时候定时器开始计时？
e. 什么时候定时器停止计时且复位？
f. 当输入端 LS1 首先闭合后，哪个梯级为真？哪个梯级为假？
g. 当输入端 LS1 首先闭合后，陈述每个输出端的状态（on 或 off）。
h. 当定时器的累加值等于预置值，哪个梯级为真？哪个梯级为假？
i. 当定时器的累加值等于预置值，陈述每个输出端的状态（on 或 off）。
j. 假设梯级 1 得电后，经过 5s 失电。当电源恢复时，计数器的累加值为多少？

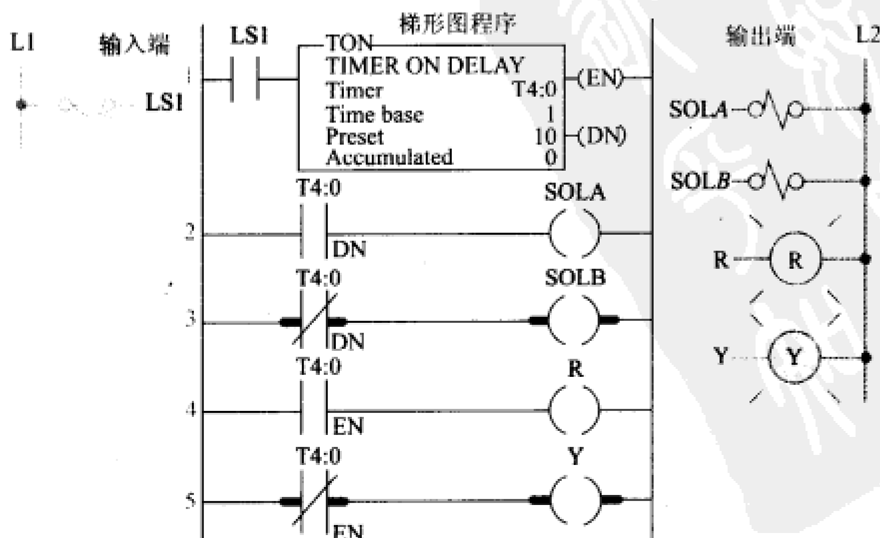


图　7-32

4. 研究图 7-33 所示的梯形图程序，并且回答以下问题：
   a. 程序中用了什么类型的定时器？　b. 延时区间的长度为多少？
   c. 什么时候定时器开始计时？　d. 什么时候定时器复位？
   e. 什么时候梯级 3 为真？　f. 什么时候梯级 5 为真？
   g. 什么时候输出端 PL4 通电？
   h. 假设累加时间值达到 020，然后系统失电。当电源恢复时，累加时间值为多少？

[196] i. 如果输入端 PB1 和 PB2 同时为真，将出现什么状况？

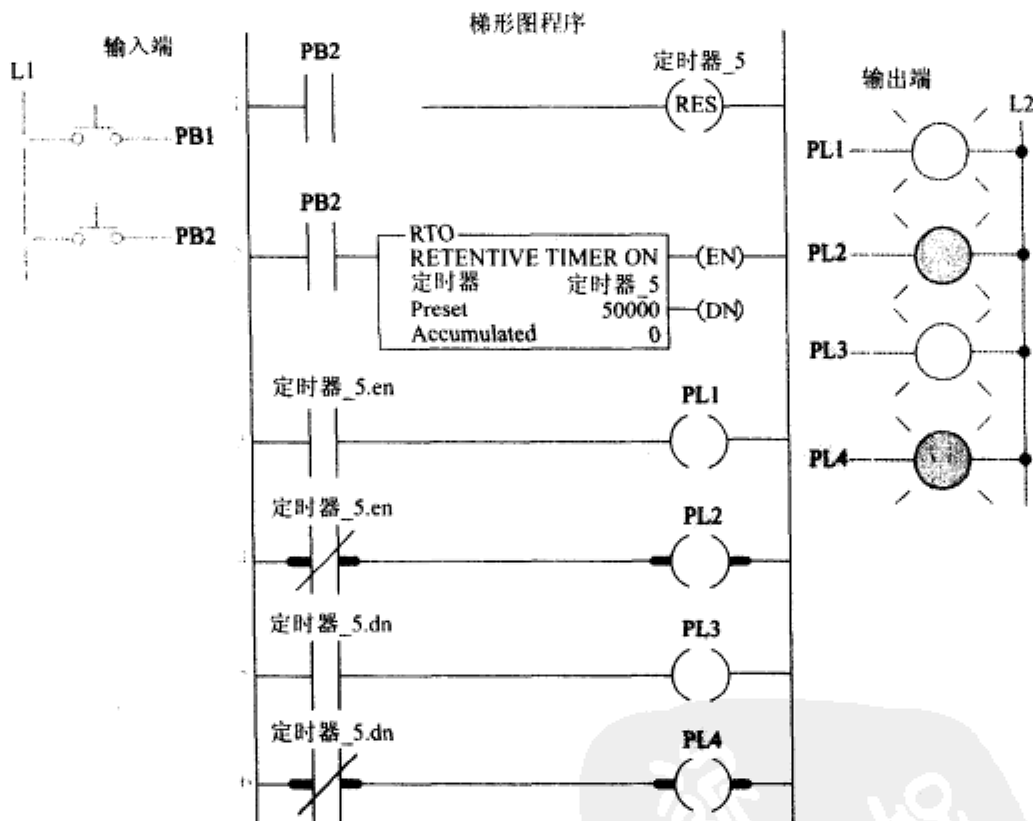


图　7-33

5. 研究图 7-34 所示的梯形图程序，并且回答以下问题：
   a. 两个定时器实现互联的目的是什么？　b. 在输出端 PL 通电前将耗费多少时间？
   c. 定时器 T4:2 要开始计时必须满足哪两个条件？
   d. 假设输出端 PL 导通后系统失电。当电源恢复后，输出端将会是什么状态？
   e. 当输入端 PB2 导通，将出现什么情况？

[197] f. 当输入端 PB1 导通，在梯级 3 为真之前将耗费多少累加时间？

6. 有一台不断进行周期性开和闭的机器。为了维护，需要记录它的总运行时间。哪种定时器能满足要求？
7. 当开关 S1 打开 15s 后，输出端 PL 导通，指示灯亮。按照条件写出梯形图程序。
8. 研究图 7-35 所示的通延时定时器的梯形图程序，按照以下各状态，确定定时器是否复位、正在计时、完成计时或状态不存在。
   a. 输入端为真，EN 为 1，TT 为 1 且 DN 为 0。
   b. 输入端为真，EN 为 1，TT 为 1 且 DN 为 1。

198

c. 输入端为假，EN 为 0，TT 为 0 且 DN 为 0。
d. 输入端为真，EN 为 1，TT 为 0 且 DN 为 1。

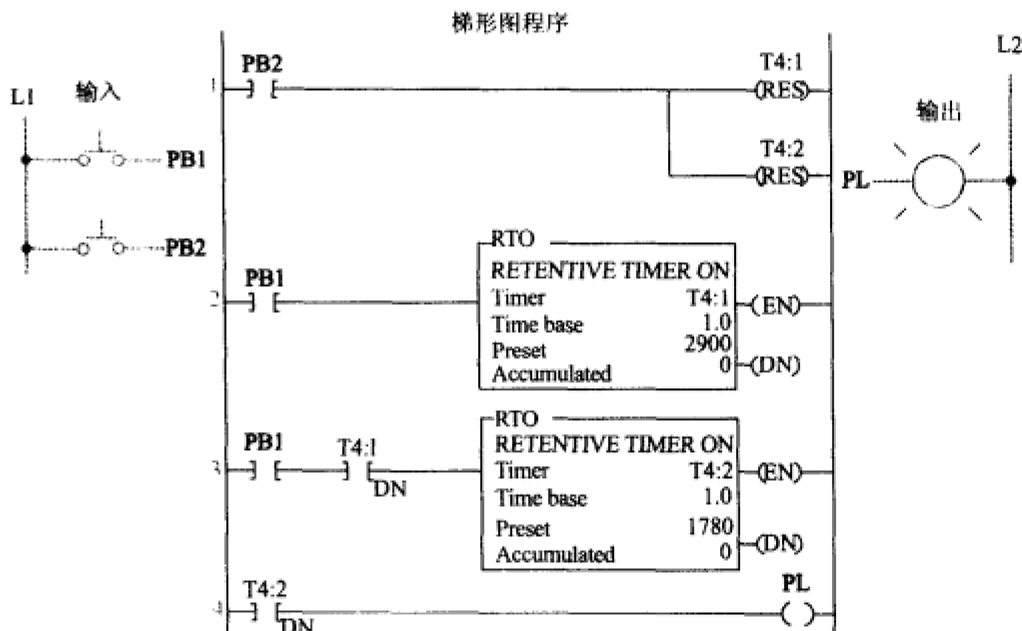


图　7-34

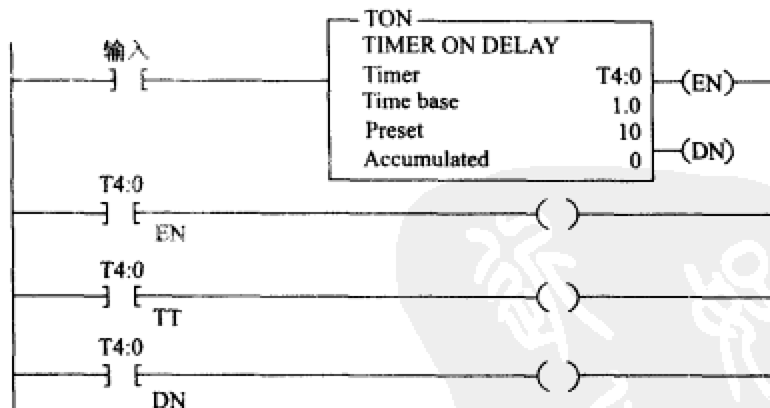


图　7-35

9. 研究图 7-36 所示的通延时定时器的逻辑梯形图程序，按照以下各状态，确定定时器是否复位、正在计时、完成计时或状态不存在。
   a. 输入端为真，EN 为 0，TT 为 0 且 DN 为 1。　b. 输入端为真，EN 为 1，TT 为 1 且 DN 为 1。
   c. 输入端为真，EN 为 1，TT 为 0 且 DN 为 1。　d. 输入端为假，EN 为 0，TT 为 1 且 DN 为 1。
   e. 输入端为假，EN 为 0，TT 为 0 且 DN 为 0。
10. 针对"反钳制电路"编写程序，除非两只手都放在手掌启动按钮上，否则电路将禁止冲床螺线管的运行。两个按钮必须在 0.5s 之内同时被按下。电路也不允许操作员钳制其中一个按钮而只按一个按钮。（提示：当按下任意一个按钮，开始计时 0.5s。如果两个按钮都没有按下，禁止螺线管运行。）

199

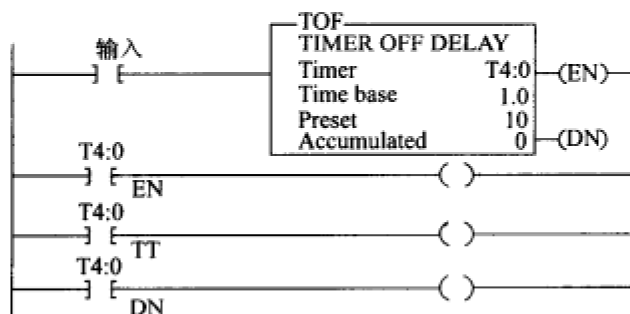


图 7-36

11. 对图 7-29 的交通控制程序进行修改，当两个方向都亮红灯时，有 3s 的延时区间。
12. 根据图 7-37 所示的执行过程编写程序。操作顺序如下：
    - ❑ 利用常开启动按钮使过程开始，利用常闭停止按钮使过程停止。
    - ❑ 当按下启动按钮，螺线管 *A* 得电，开始向容器中注入液体。
    - ❑ 随着液体注入容器，低位传感器开关闭合。
    - ❑ 当液体将容器注满，高位传感器开关闭合。
    - ❑ 螺线管失电。
    - ❑ 搅拌电动机自动启动，开始搅拌液体，此过程持续 3min。
    - ❑ 当搅拌电动机停止，螺线管 *B* 得电，液体开始流出容器。
    - ❑ 当容器中的液体全部流出，低位传感器开关断开，螺线管 *B* 失电。
    - ❑ 按下启动按钮重复以上顺序。

200

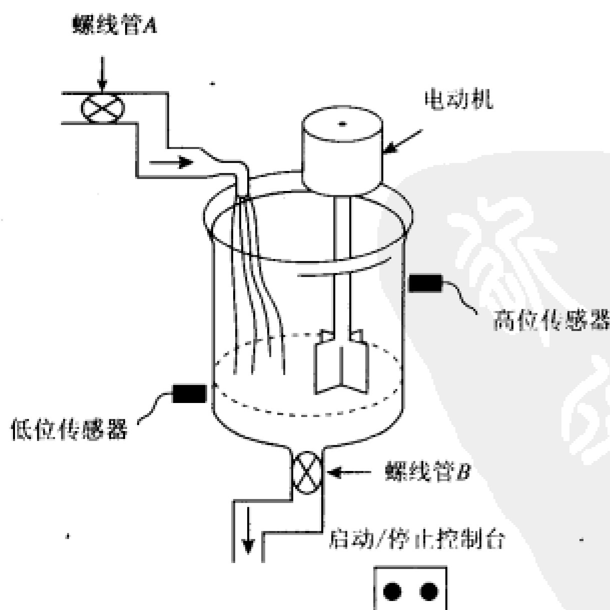


图 7-37

13. 当楼房内的灯断电时，出口指示灯还要再亮 2min，出口指示灯熄灭后，停车场指示灯继续再亮 3min。对以上过程编写程序。
14. 编写程序实现尾灯顺序运行系统的模拟。轿车两边各有三个独立的尾灯。通过操作左右信号开关，使得每组灯都可以独立工作。在每个灯接通之间有 1s 的延时，且当关闭所有灯时

也有 1s 的延时。当两个开关都闭合时，系统不会运行。尽可能少的使用定时器。运行顺序如下所示：

- 接通开关。
- 灯 1 亮。
- 1s 后，灯 2 亮。
- 1s 后，灯 3 亮。
- 灯 3 持续 1s。
- 1s 后，所有灯熄灭。
- 当接通开关，系统重复运行。

第 8 章　程序计数器

学习目标：

- ❑ 列举并且描述 PLC 计数器指令的功能。
- ❑ 描述过渡触点或一次触点的工作原理。
- ❑ 分析说明典型 PLC 计数器的梯形图程序。
- ❑ 运用 PLC 计时器指令和相关电路实现对系统的控制。
- ❑ 运用计数器和定时器的组合实现对系统的控制。

大多数 PLC 都包含加计数器指令和减计数器指令，这两种指令功能相似。本章对梯形图中的计数器指令及其功能作了解释。典型的 PLC 计数器指令应用实例包括：在过程中直接计数、使用两个计数器对两个计数值求和或使用两个计数器求两个计数值之差。

8.1　计数器指令

程序计数器具有同机械式计数器相似的功能。图 8-1 显示了一种简单的机械式计数器的构造。执行层每移动一位，计数器就增加一位数；然后执行层自动返回到初始位置。通过位于装置旁边的按钮可以实现清零。

电子计数器（如图 8-2 所示）能够实现加计数、减计数或实现加减计数。虽然在工业中主要使用的是加计数器，但仍有大量应用要求使用减计数器或加/减计数器。每种型号的 PLC 都在其指令集中提供了一些计数器指令。图 8-3 举例说明了计数器的典型应用。计数器的一般应用包括追踪部件经过一点的次数，和决定规定动作的出现次数。被 PLC 的计数器功能替代的传统计数器包括机械式、电气式及电子式。

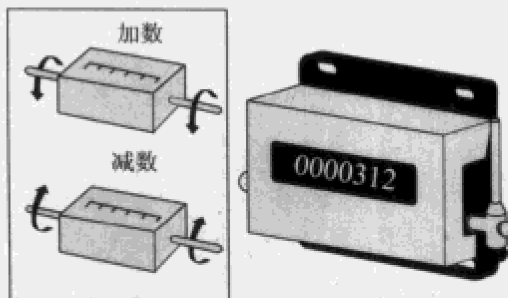


图 8-1　机械式计数器

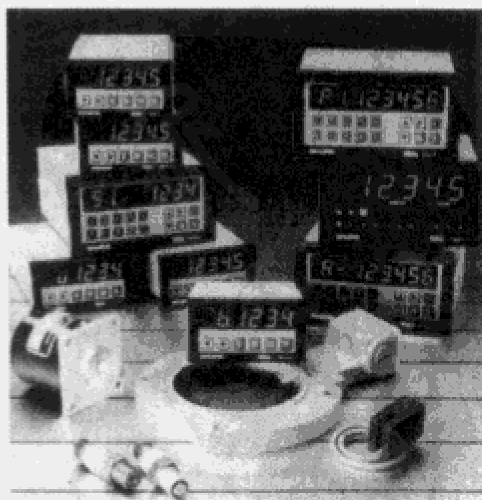


图 8-2　电子计数器（得到位于美国伊利诺伊州 Gurnee 市的 Dynapar 公司许可）

计数器和定时器很相似，但是计数器工作时不会依靠内部时钟，它的工作由外部条件或程序源决定。用 PLC 的梯形图程序来表示计数器的方法有线圈式和组块式两种。线圈式的编程方法如图 8-4 所示。计数器被分配一个地址并且被标识为计数器。在计数器指令

中，同样包括计数器的预置值和计数器的当前累加计数值。当计数器梯级每次由假变为真时，加计数器将它的累加值增 1。当累加值等于预置值时，输出端得电且计数器的输出端闭合。通过程序，计数器的触点能够根据需要使用任意多次，就像 NO 触点或 NC 触点的使用一样。

203
204

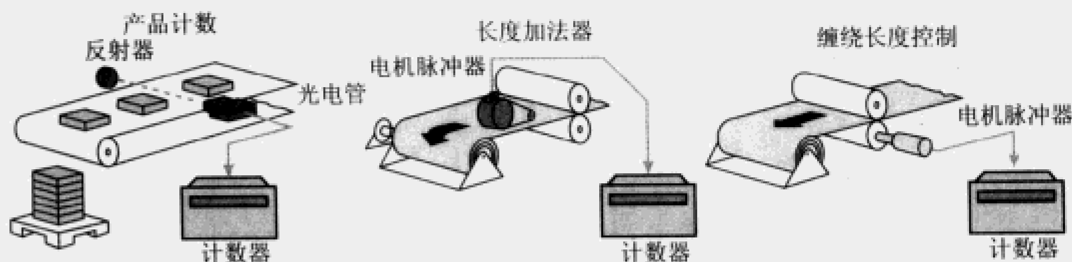


图 8-3　计数器的运行

在计数器指令中也有使计数器复位的计数器复位指令。加计数器通常被置零；减计数器可以被置零或置为其他预置值。有些厂家将复位指令设计成通用计数器指令的一部分，而有些厂家将一个独立的指令用作计数器的复位。图 8-5 显示了具有独立复位指令的一类线圈式计数器指令。在编程时，计数器复位线圈（CTR）被赋予和待复位计数器（CTU）相同的参考地址。当 CTR 梯级的状态为真时，复位指令被激活。

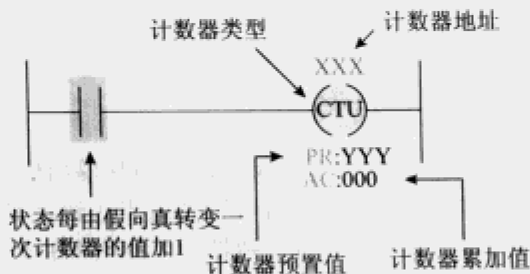


图 8-4　线圈式计数器指令

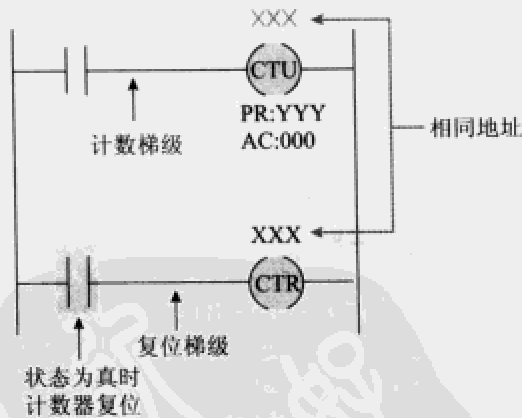


图 8-5　线圈式计数器和复位指令

第二种计数器的表示形式为组块式。图 8-6 举例说明了一类组块式计数器指令。组块式指令标示了计数器的类型（加数或减数），以及计数器的预置值和累加值或当前值。有两个与计数器相关的输入条件，即计数和复位。所有的 PLC 计数器都是在输入信号上升沿到来时运行或计数。当计数输入端从断开状态转为闭合状态时，计数器将进行加数或减数。在输入信号的下降沿，或输入端的状态从闭合转为断开状态时，计数器不会运行。

有些厂家需要利用复位梯级或一条线路为真时来对计数器进行复位，而有些厂家需要当线路为假来对计数器复位。由于这个原因，在尝试任何计数器电路编程前应查阅相关的 PLC 操作手册。

多数 PLC 的计数器通常为保持型，即在通电的情况下，当处理器运行停止后，计数器中的值都会储存在计数器中。然而当电源恢复时，复位条件被激活，计数器的值将被复位。

205

可以将 PLC 计数器设计为从预置值开始加数或减数。当计数器梯级每导通一次，加计数器将使其值加 1，而对于减计数器，其值将减 1。操作处理程序中的一些事件，将导

致梯级状态的改变，如传感器接收到信号或限位开关动作。PLC的预置值可以被设置，也可以放入存储单元作为程序判定的结果。图8-7举例说明了加计数器和减计数器的计数次序。计数器显示的值称为*累加值*。计数器能实现加数或减数取决于它的类型，当累加值等于或大于预置值时，将计加数，此时将产生一个输出信号。通常计数器的复位都是将计数器的累加值复位到一个预定的值。

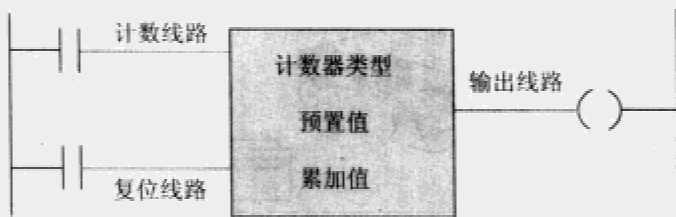
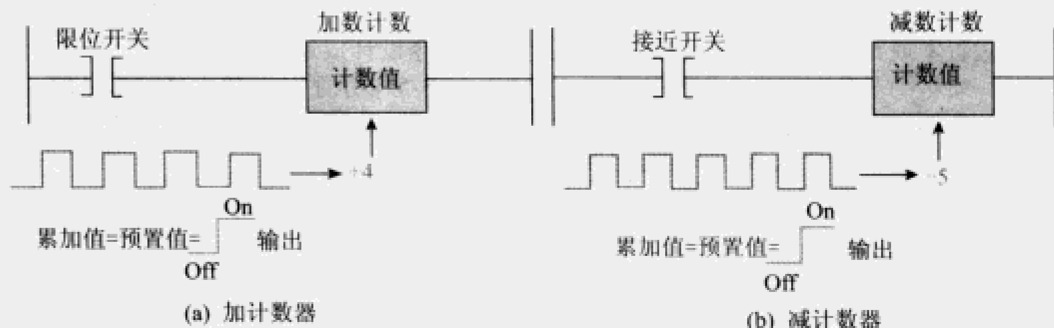


图8-6 组块式计数器指令



(a) 加计数器

(b) 减计数器

图8-7 计数器计数次序

8.2 加计数器

加计数器作为一条输出指令，其功能为当指令状态由假变为真时，将增加其累加值。因此当输入指令由假变为真时，加计数器可以对其变换的次数进行计数，然后当达到要求的计数或变换次数后触发一个事件。被计数的事件每出现一次，加计数器指令的值将加1。图8-8显示了一个简单加计数器程序和时序图。设计此程序的目的是当累加值为7后，红色信号灯亮且绿色信号灯熄灭。计数器所计数的从断开到导通的过渡脉冲由操作按钮PB1提供。计数器的预置值设为7。梯级1的状态由假变为真转变一次，计数器的累加值将加1。在经过7个脉冲或计数后，预置值等于累加值，输出端DN导通。然后梯级2变为真且输出端O:2/0导通，红色指示灯亮。同时，梯级3变为假且输出端O:2/1失电，绿色指示灯熄灭。通过闭合按钮PB2，梯级4为真且累加值置零，计数器实现复位。当梯级4再次变为假时，计数重新开始计数。

206 每条Allen-Bradley PLC-5和SLC-500计数器指令，占据C5计数器数据文件中三个字的存储单元（如图8-9所示）。这三个数据字分别为控制字，预置字和累加字。在这三个数据字中，每个数据字都共享相同的基地址，该基地址为计数器本身的地址。控制字使用的状态控制位包括以下几个：

- **加数（CU）使能位**

加数使能位用于加计数器中，且当加计数器指令为真时，其值为真。如加计数器指令

207 为假，CU位为假。

❑ **减数（CD）使能位**

减数使能位用于减计数器中，且当减计数器指令为真时，其值为真。如减计数器指令为假，CU 位为假。

❑ **完成（DN）位**

不论是加计数器还是减计数器，当计数器的累加值等于或大于预置值，完成位为真。

❑ **溢出（OV）位**

当计数器的计数值超过其最大值 32 767 时，溢出位为真。当加计数器状态逐次由假为真的变换时，在溢出后的下一个值到来时，计数器将跳转到 -32 768，且一直计数到 0。

❑ **下溢（UN）位**

当计数器的计数值低于 -32 768 时，下溢位将为真。当减计数器梯级逐次由假为真进行变换时，在溢出后的下一个值到来时，计数器将跳转到 +32 767，且一直计数到 0。

❑ **累加更新（UA）位**

累加更新位只与外部 HSC（高速计数器）相关。

预置值（PRE）字指定了一个值，在完成位的状态改变以前，计数器必须计数到这个值。预置值为计数器的设定值，其范围从 -32 768 到 +32 767。数值以二进制的形式储存，且任何负数以第二种二进制补码形式储存。

累加值（ACC）字为基于梯级由假变为真的次数的当前计数值。当加计数器指令状态由假变为真时，累加值增加；当减计数器指令状态由假变为真时，累加值减少。累加值具有同预置值一样的变换范围：从 -32 768 到 +32 767。计数器的累加值超过预置值后还会继续计数，而不会像定时器一样停止在预置值。

图 8-10 显示了在 Allen-Bradley PLC-5 和 SLC-500 控制器指令集中，加计数器及其状态位使用的例子。计数器的地址从 C5:0 开始，到 C5:999 结束。接收的信息包括： [208]

❑ **计数器编号**

计数器编号来自计数器文件。如例中所示，计数器编号为 C5:0，表示计数器 0 在计数器文件 5 中。在每个文件中可以有最多 1000 个计数器，其编号从 0 一直到 999。例中计数器的地址不能用于其他加计数器。

❑ **预置值**

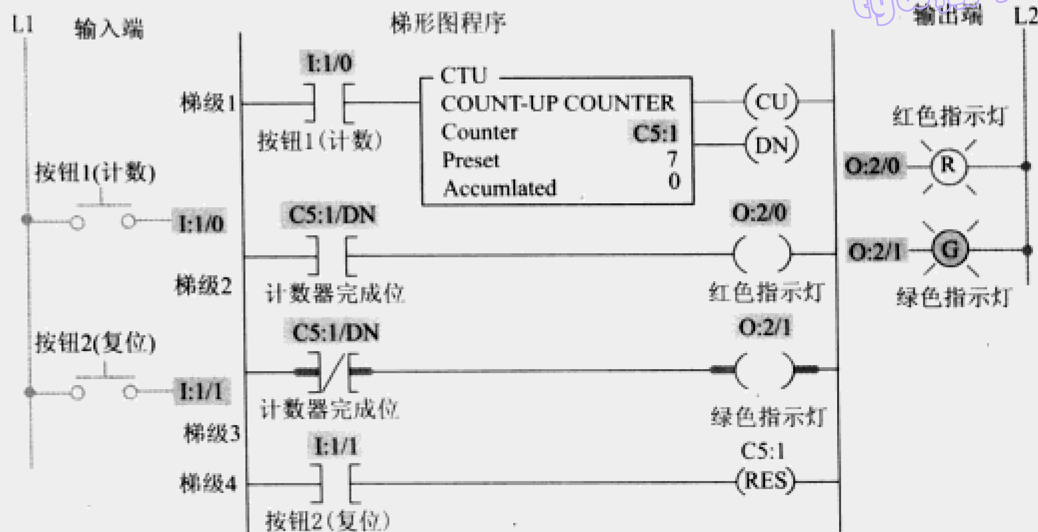
预置值的范围从 -32 768 到 +32 767。如例中所示，预置值为 10。

❑ **累加值**

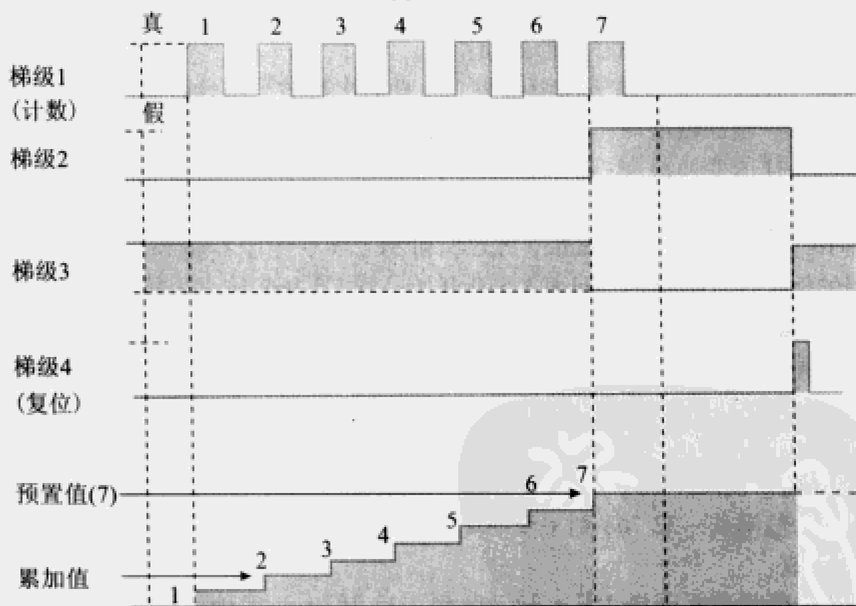
累加值的变化范围同样为从 -32 768 到 +32 767。一般地，如此例，输入到累加字中的值为 0。不管输入什么值，复位指令都将累加值复位到 0。

虽然 PLC-5 和 SLC-500 与 ControlLogix 控制器的计数器指令的功能相同，但计数器的地址、最大的预置值及累加值不同。在 PLC-5 和 SLC-500 中，计数器的地址为数据表地址；而在 ControlLogix 控制器中，其地址为计数器数据类型中预定义结构体类型（如图 8-11 所示）。在 PLC-5 和 SLC-500 中，预置值与累加值的最大值为 32 767，最小值为 -32 768；对于 ControlLogix 控制器中，其最大值为 2 147 438 647，最小值为 -2 147 438 648。

图 8-12 显示了 RSLogix 工具栏中定时器/计数器的功能单列表。当选中此列表，将会出现几种定时器和计数器指令。前面三种为第 7 章介绍过的定时器指令。从左边开始接着的两条指令为加计数器指令（CTU）和减计数器（CTD）指令。在 CTU 和 CTD 指令的右边为复位（RES）指令，其既用于计数器也用于定时器。 [209]



(a) 程序



(b) 时序图

图 8-8 简单加计数器程序

计数器地址																	
C5:N	位	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
C5:N.0	字	CU	CD	DN	OV	UN	UA	内部调用(不可编址)									
C5:N.1	字1	预置值															
C5:N.2	字2	累加值															

图 8-9 C5 计数器数据文件

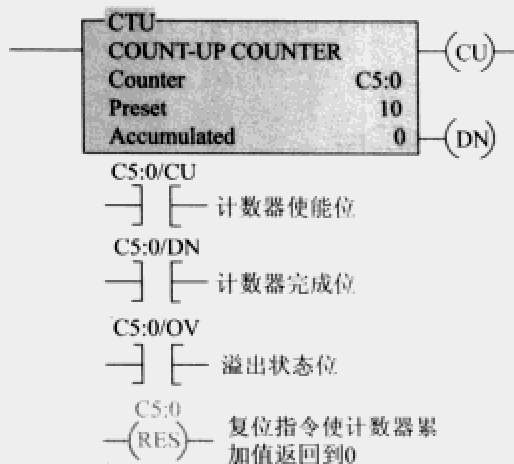


图 8-10 加计数器指令举例

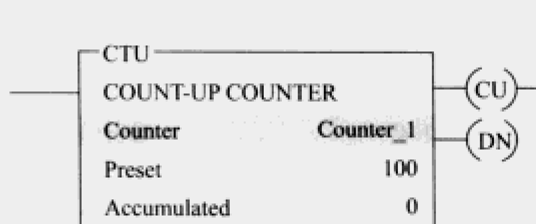
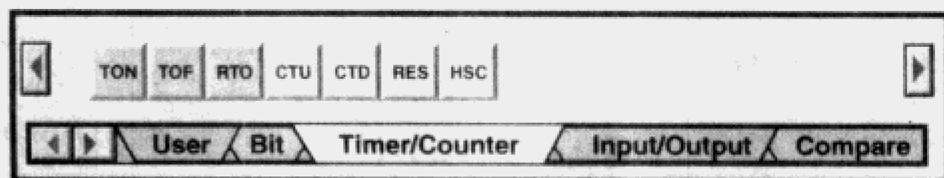


图 8-11 ControlLogix 控制器的加计数器指令



命 令	全 称	描 述
CTU	加计数器	每当状态由假向真转变时，累加值增加，当电源循环供电时，累加值保持不变
CTD	减计数器	每当状态由假向真转变时，累加值减少，当电源循环供电时，累加值保持不变
HSC	高速计数器	对控制器固定的高速输入的脉冲进行计数

图 8-12 RSLogix 工具栏中定时器/计数器功能单列表

图 8-13 显示了使用了三个 PLC 加计数器的部件计数程序。计数器 C5:2 对装配线中最后包装完成的部件个数进行计数。每次包装必须包含 10 个部件。当检测到 10 个部件时，计数器 C5:1 置位 B3/1，开始对盒子进行封装。计数器 C5:3 对一天中完成的总包装盒数进行计数。（每天包装的最大数目为 300 盒。）用一个按钮来实现部件数和包装盒数的清零。

图 8-14 显示了常用的对计数器自动清零或复位的一次触发电路或过渡触发电路的程序。当收到触发信号，电路产生一个输出脉冲，脉冲持续一个程序扫描周期后消失。一次触发电路可以通过瞬时信号触发或通过接收一个信号来触发，触发后电路输出状态将保持一定时间。无论使用什么信号，一次触发电路都是通过输入信号的上升沿（断开到导通）来触发。输出状态将在一个扫描周期后消失。由于一次触发电路的输出状态只保持在一个扫描周期内，因此它非常适用于对计数器和定时器的复位。

210

一些 PLC 除了提供标准的 NO 和 NC 触点指令外，还提供了过渡触点指令或一次触发指令。当触发信号的上升沿（断开到导通）到来时，过渡触点（如图 8-15a 所示）便产生一个一次触发脉冲。无论触发信号在什么时候从断开到导通，过渡触点将在一个扫描周期时间内处于闭合状态。即使触发信号可以一直保持，过渡触点也只允许逻辑接通一个扫描周期，然后断开。除了当触发信号为导通转为关断状态时下降沿型过渡触点允许逻辑信号

持续一个扫描周期以外，下降沿型过渡触点（如图8-15b所示）的操作同上升沿型过渡触点一样。

(a) 过程示意图

(b) 梯形图程序

图8-13 部件计数程序

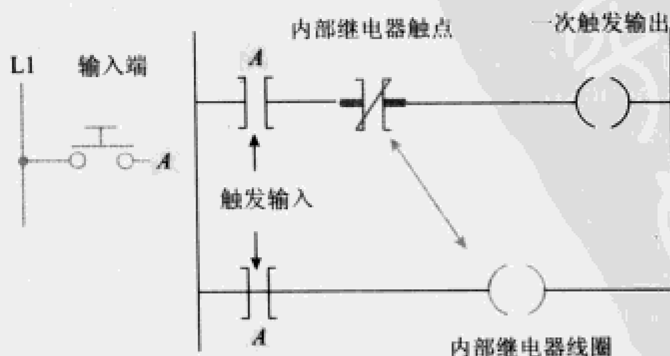


图8-14 一次触发电路或过渡电路程序

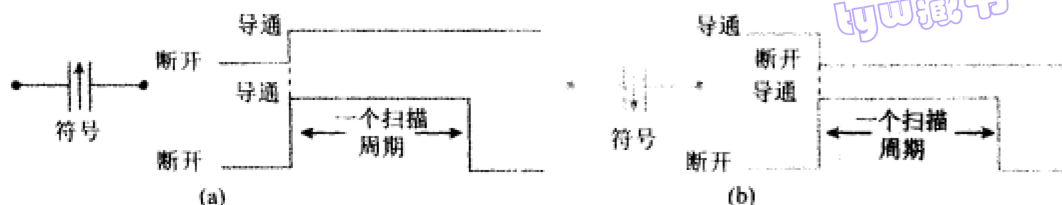
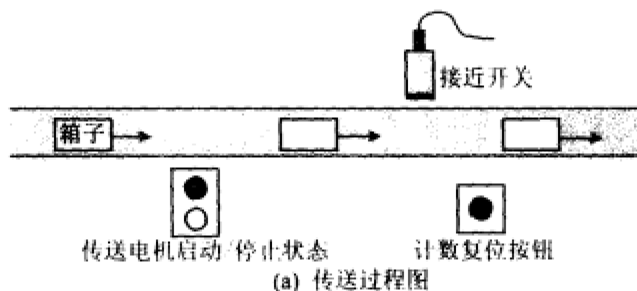
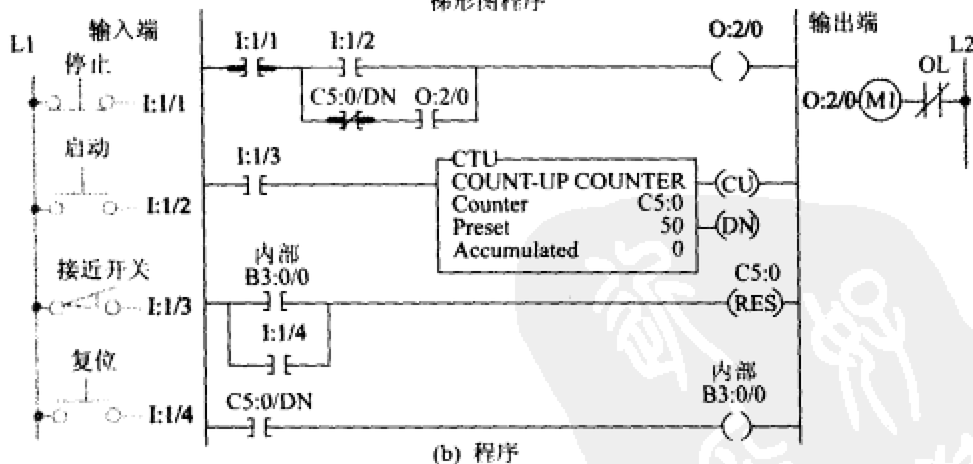


图 8-15 两种类型的过渡触点

图 8-16 所示的传送机 PLC 程序，是加计数器结合一次触发复位电路的应用。计数器对通过传送机的箱子进行计数，当箱子总数达到 50，传送机将自动停止运行。然后通过卡车将这些固定数目的货物运走。根据不同的生产线，数目可以更改。接近开关用于检测箱子的通过。 211



(a) 传送过程图



(b) 程序

图 8-16 传送机程序

工作顺序如下：

(1) 按下启动按钮，传送机开始工作。

(2) 箱子通过接近开关，计数器累加值增加。

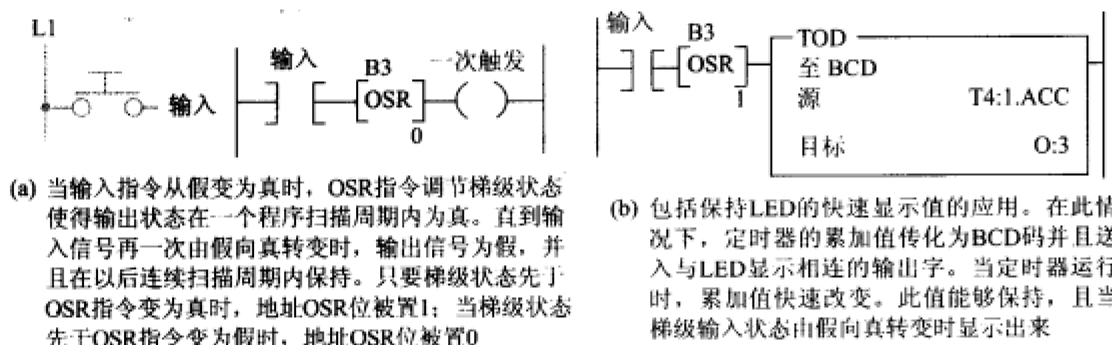
(3) 累加值到达 50 后，传送机自动停止运行且计数器累加值清零。

(4) 在任何时候，传送机都可以人为启动或停止，而不需要累加计数的复位。

(5) 在任何时候计数器的累加计数都可以通过复位按钮实现人为复位。

如图 8-17 所示的 Allen-Bradley 上升沿一次触发（OSR）指令为保持型输入指令，它用于产生一个动作。当一个动作的开始必须基于梯级状态由假变为真，而不是基于梯级结果的状态时，需要用 OSR 指令。分配给 OSR 指令的地址不是程序中规定的一次触发的地 212

址。地址允许 OSR 指令记忆梯级的初始状态。通过“一次触发”程序，可以规定 OSR 指令后的输出指令。



(a) 当输入指令从假变为真时，OSR指令调节梯级状态使得输出状态在一个程序扫描周期内为真。直到输入信号再一次由假向真转变时，输出信号为假，并且在以后连续扫描周期内保持。只要梯级状态先于OSR指令变为真时，地址OSR位被置1；当梯级状态先于OSR指令变为假时，地址OSR位被置0

(b) 包括保持LED的快速显示值的应用。在此情况下，定时器的累加值传化为BCD码并且送入与LED显示相连的输出字。当定时器运行时，累加值快速改变。此值能够保持，且当梯级输入状态由假向真转变时显示出来

图 8-17 Allen-Bradley 一次触发（OSR）指令

图 8-18 为 PLC 的报警监视程序。在此程序中运用了一个加计数器和第 7 章介绍的定时振荡电路。报警监视器的操作如下：

（1）通过闭合液位开关 LS1 触发报警信号。

（2）当报警条件被触发且没有被确认时，指示灯开始闪烁，即使同时报警条件又被清除，指示灯仍然闪烁。

（3）通过闭合选择开关 SS1，报警信号得到确认。

213 （4）当报警信号的触发条件仍然存在，但是已经被确认时，指示灯将工作在稳定状态。

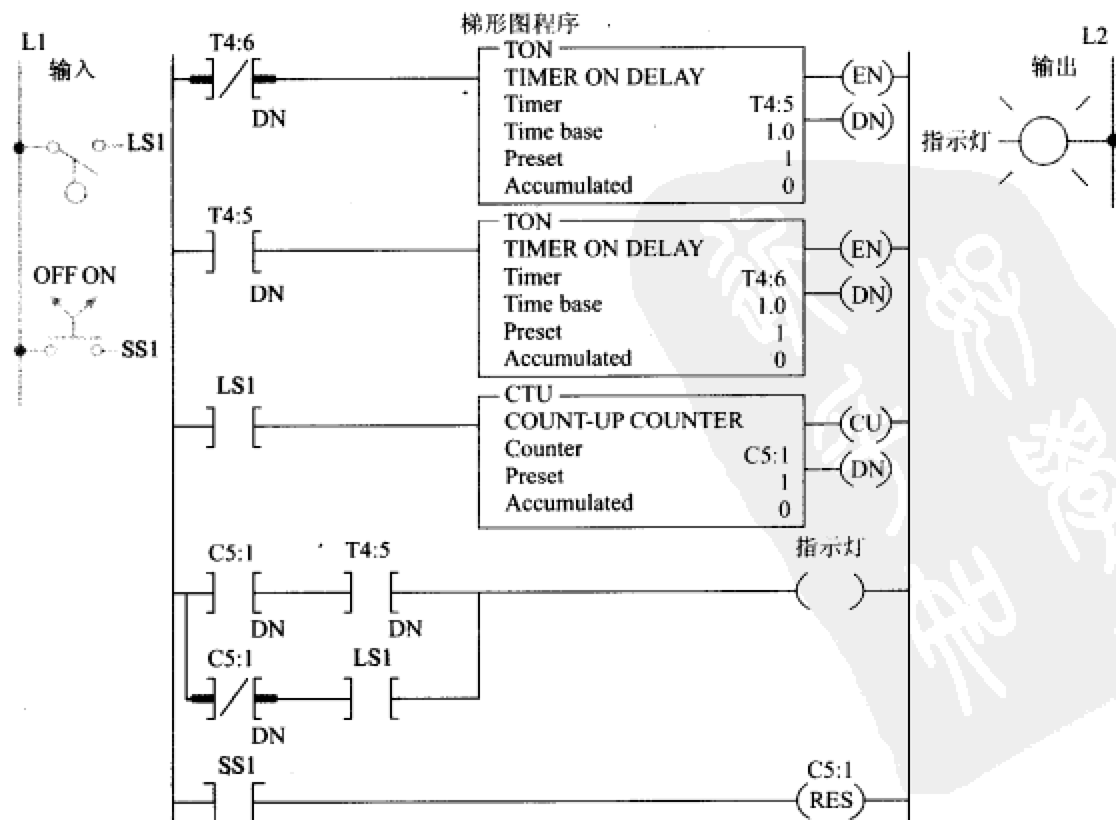


图 8-18 报警监视器程序

8.3 减计数器

每当被计数的事件出现一次，减计数器指令将使计数值减1。被计数的事件每次出现，累加值都会减少。通常将减计数器和加计数器结合起来形成加/减计数器。图8-19显示了一般的组块式加/减计数器的程序及时序图。加数和减数的输入端都是独立的。假设计数器的预置值为3且累加值为0，当三个脉冲通过加数输入端（PB1）后，输出端指示灯由关断变为导通。在累加值大于预置值的情况下，特殊PLC的计数器将一直对计数器的值进行跟踪。当额外的三个脉冲通过加数输入端（PB1）后，累加值为6，但输出端状态不发生改变。如果现在有四个脉冲通过减数输入端（PB2），累加值将为2×（6-4）。结果，累加值减小到预置值以下，且输出指示灯从导通变为关断。在任何时刻向复位输入端（PB3）送入一个脉冲将使累加值清零，并且关断输出端的指示灯。

214

不是所有的计数器指令都以同样的方式计数。一些加计数器只计数到其预置值，而忽略掉大于预置值的计数。而另外一些加计数器会对超过计数器预置值的计数继续保持追踪。相反，一些减计数器只是简单地计数到0，且不会进一步的计数。而另一些减计数器可以在计数到0之后，又从PLC计数器指令允许的最大预置值开始向下计数。例如，最大计数预置值为999的PLC加/减计数器，可按如下方式进行加计数：997，998，999，000，001，002，等等。同样的计数器可以按以下方式进行减数计数：002，001，000，999，998，997，等等。

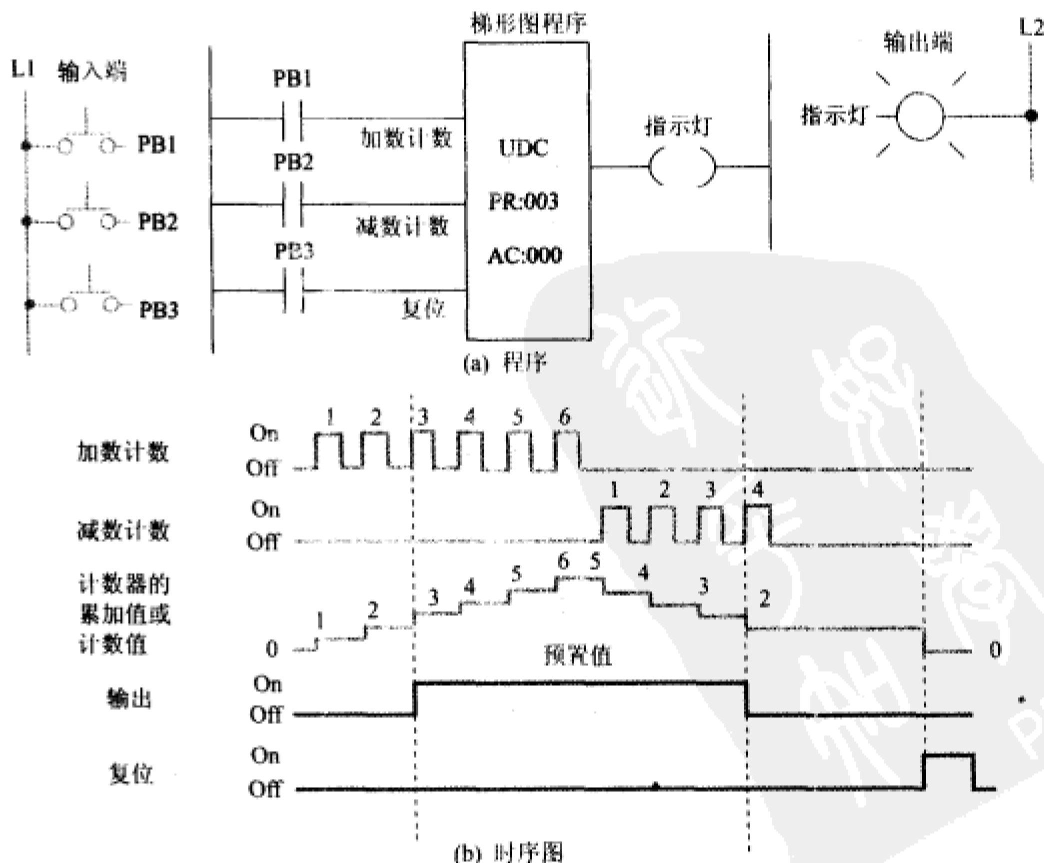


图8-19　一般的加/减计数器程序

对于加/减计数器的一个典型的应用就是对进入或离开停车库的汽车数目进行计数。当汽车进入车库时，触发加计数器指令且使累加值增加1。相反，当汽车离开时，触发减计数器指令且累加值减少1。由于加计数器和减计数器都具有相同的地址，因此它们的累加值相同。当累加值等于预置值时，计数器输出端将输出停车已满的信号。图8-20显示了用于实现该电路的典型的PLC程序。在电路中提供了一个复位按钮用于使累加值复位。

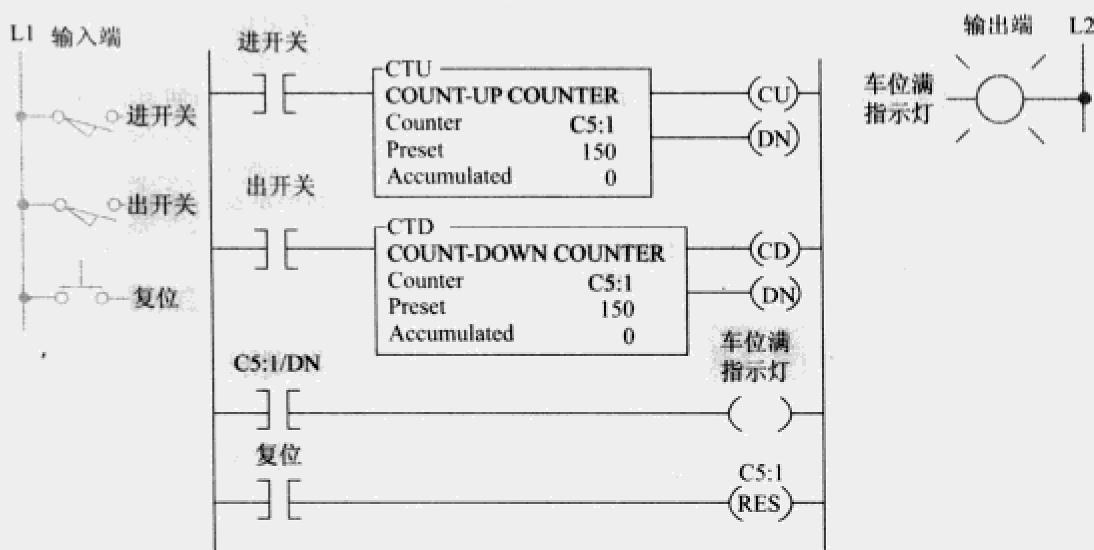


图8-20 车库中停车计数

图8-21显示了Allen-Bradley PLC-5和SLC-500控制器指令集中的减计数器的使用例子。指令中所输入的信息和加计数器指令中的信息相同。

CTD指令每改变一次状态，它的累加值将减1。当累加值等于或大于预置值时，完成位将置1。CTD指令需要RES指令来对它的累加值和位状态进行复位。由于累加值清零，所以当CTD指令状态改变时，其值为负。如果减计数器的预置值为正，则当累加值达到0时，其完成位将被置位。然后，计数将朝负方向进行，累加值将不可能达到其预置值，也不能将完成位置1。然而，当输入一个负的预置值时，且当累加值小于预置值时，完成位将被置1。



图8-21 减计数器指令样例

图8-22显示了加/减计数器程序。按下按钮PB1，计数器的累加值增加；按下按钮PB2，计数器的累加值将减小。注意加计数器指令、减计数器指令和复位指令都被赋予相同的地址。所有这三条指令都将看作在计数器文件中是处于同一个地址。当输入端A由假变为真时，累加值将增加1。当输入端B由假变为真时，累加值将减少1。程序的运行可以概括如下：

- 当 CTU 指令为真时，C5:2/CU 将为真，导致输出端 A 为真。
- 当 CTD 指令为真时，C5:2/CD 将为真，导致输出端 B 为真。
- 当累加值大于或等于预置值时，C5:2/DN 将为真，导致输出端 C 为真。
- 输入端 C 为真将使两个计数器指令复位。当通过 RES 指令复位后，累加值清零且完成位被复位。

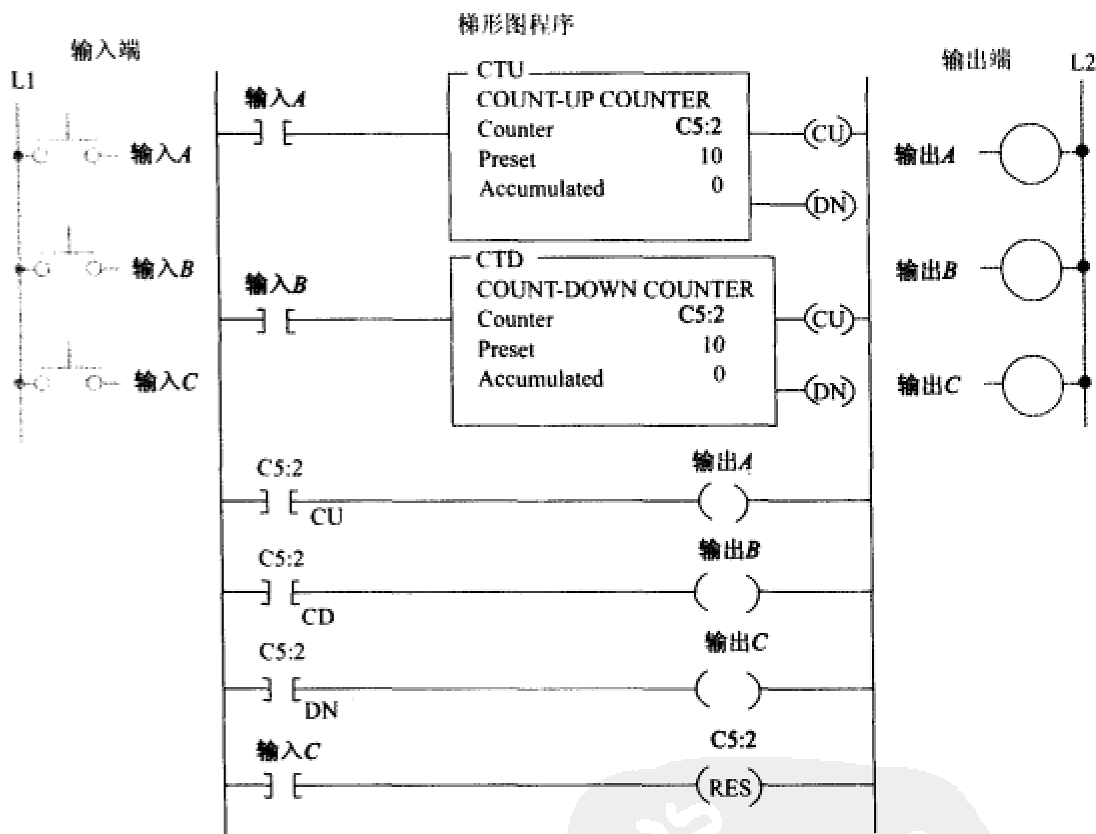


图 8-22　加/减计数器程序

图 8-23 举例说明了加/减计数器程序的工作原理。在此例中，运用计数器实现对过程中部件的连续监视。进料光电传感器对进入系统未处理的部件进行计数，而出料光电传感器对加工完成的部件进行计数。位于进料传感器和出料传感器之间的部件的数目通过计数器的累加值来表示。加数输入端的数值增加而减数输入端的数值减少。程序的运行可以概括如下：[217]

- 在启动前，系统中无任何部件且计数器被手动置零。
- 当开始运行，未加工部件通过进料传感器，每通过一个部件，计数累加值增 1。
- 完成加工后，完成的部件通过出料传感器，且每通过一个部件，计数累加值减 1。因此计数器的累加值，连续的显示出处于加工过程中的部件的数目。

计数器的预置值在此例子中不重要。计数器的输出端为导通或关断都不重要。此系统中没有使用输出端的开-关逻辑。因此可以任意地设置计数器的预置值，如设置为 50。

计数的最大速度由程序的扫描时间决定。为了得到一个可靠的计数，计数器的输入信号必须在一个扫描周期内固定。如果输入信号改变的时间小于扫描周期，计数值会因为计

数丢失而变得不可靠。当出现这种情况时，需要使用高速计数器输入端或针对高速应用而设计的独立计数器 I/O 模块。

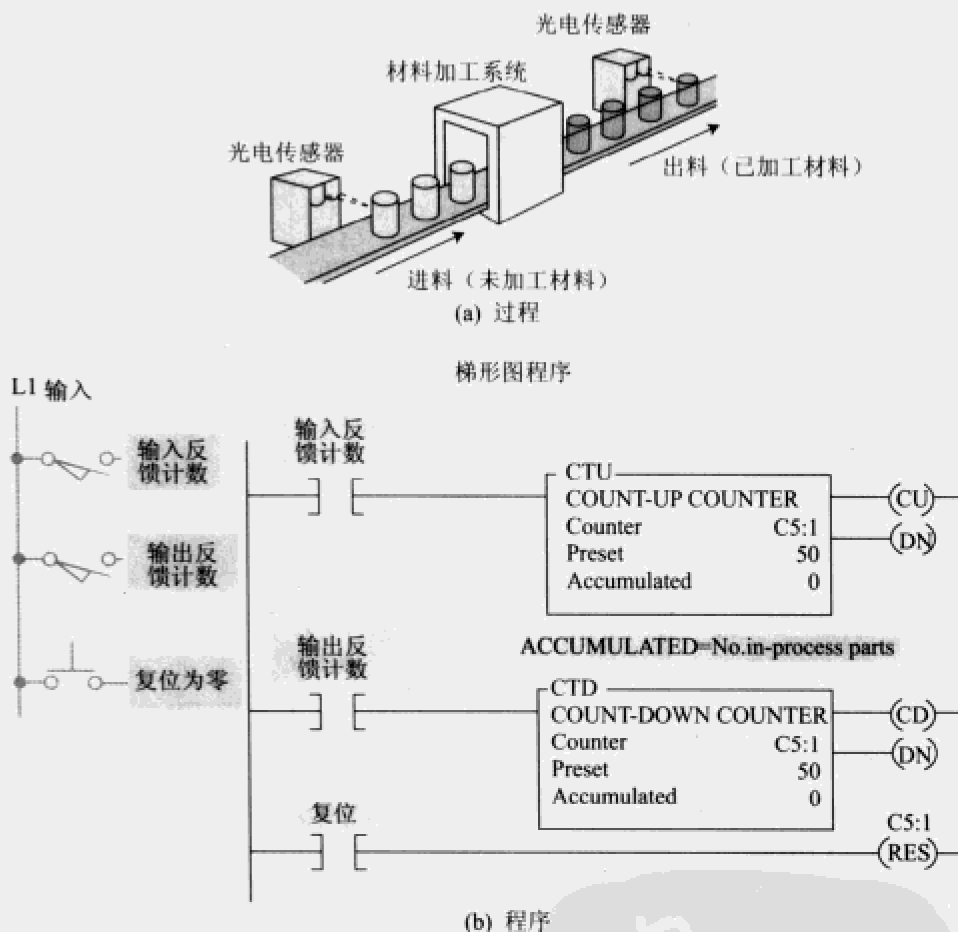


图 8-23　加/减计数器用于加工过程的连续监测系统

8.4　计数器级联

218 在一些实际应用中，有时需要对超过每条计数器指令所允许的最大计数值的事件进行计数。实现这种计数的一个方法是可以将两个计数器互联或级联。图 8-24 中的程序说明了此方法的运用。在程序中，通过编程第一个计数器的输出端为第二个计数器的输入端。两个计数器的位状态串联起来产生一个输出信号。这两个计数器允许的计数值为它们单个最大计数值的两倍。

当需要对一个极大的数进行计数时，需要使用另外一种常用的计数器级联方法。例如，如果需要计数到 250 000，只需要使用两个计数器就能实现。图 8-25 显示了怎样对两个计数器编程来达到此目的。计数器 C5:1 的预置值为 500，计数器 C5:2 的预置值为 500。当计数器 C5:1 计数到 500，其完成位对 C5:1 复位且计数器 C5:2 计数增 1。当 C5:1 的完成位由导通变为关断重复 500 次后，输出指示灯亮。因此，输出指示灯在经过 500 × 500 也就是 250 000 次计数输入后发亮。

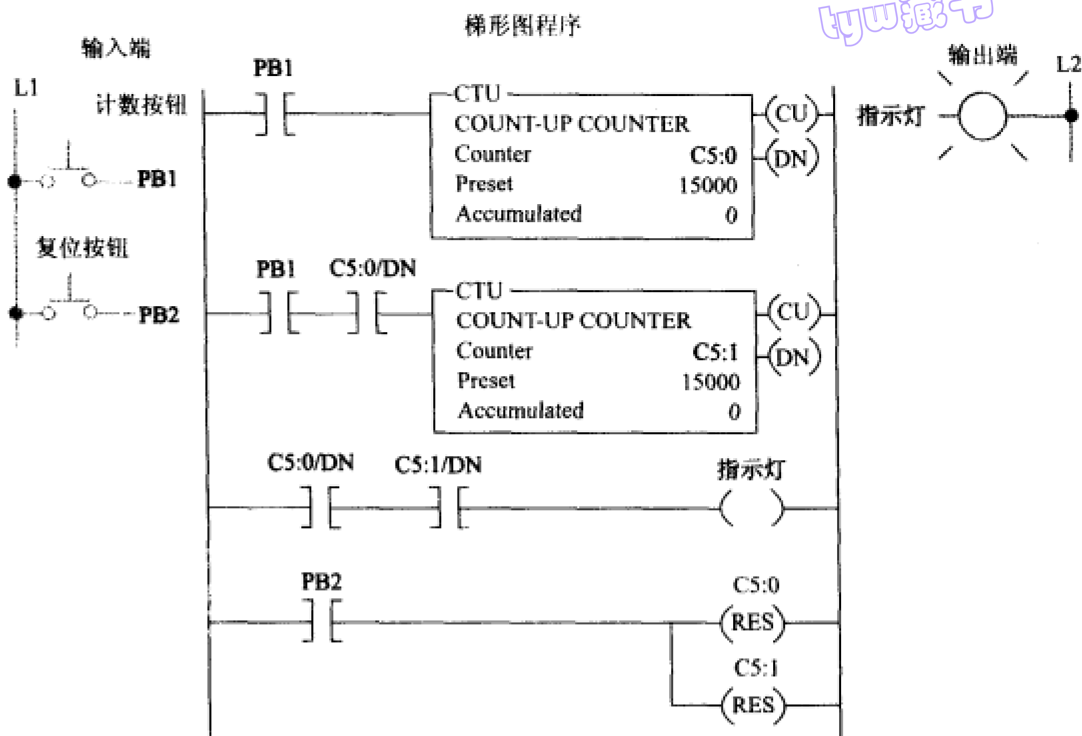


图 8-24 对超过最大计数值的数计数

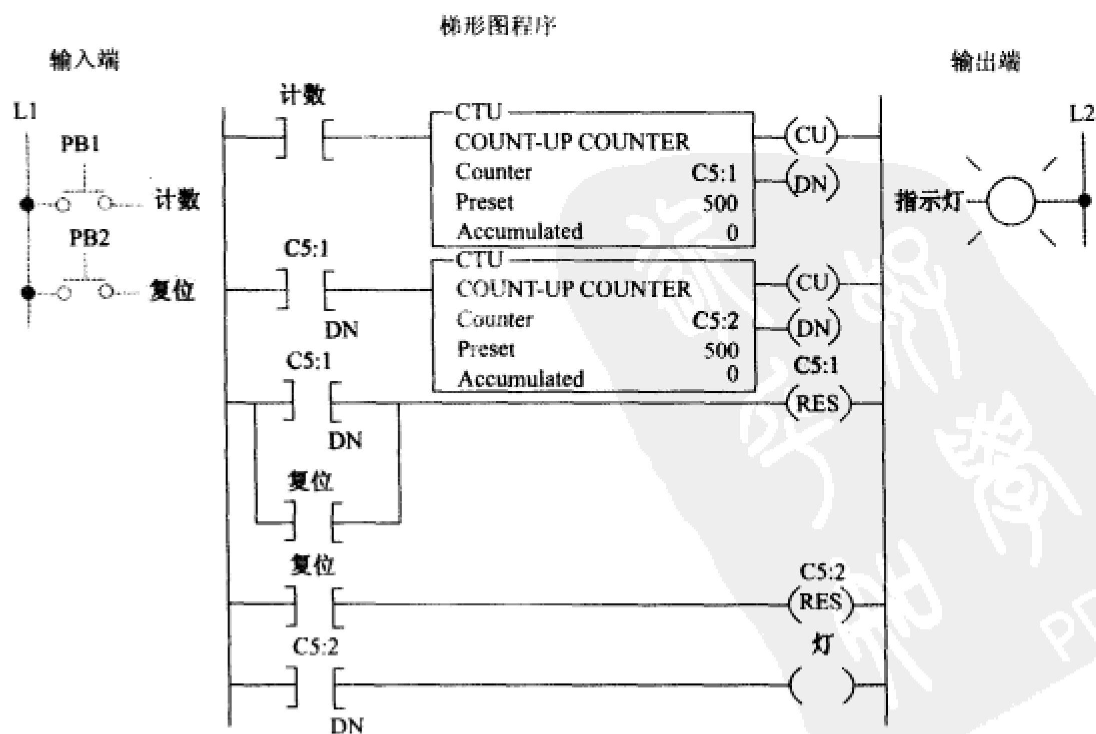


图 8-25 两个计数器级联以实现极大值的计数

有些控制系统带有一个 24 小时时钟，此时钟用于显示一天的时间或记录与运行过程 [219] 相关的数据。时钟信号作为 PLC 程序中的一部分，其原理很容易实现。其中需要使用定时器指令和计数器指令。

图 8-26 说明了一个能够产生时钟的定时 - 计数程序，此时钟以时和分为单位来显示时间。首先通过程序将 TRO 定时器指令（T4:0）的预置值设为 60s。此定时器计时以 60s 为周期，60s 计时完成后其完成位置 1。这使得梯级 001 中的加计数器（C5:0）计数增加 1。在下一个处理器扫描周期开始时，定时器被复位且开始重新计时。C5:0 计数器的预置值设为 60，且当定时器每完成一个时间延时周期，计数器的值都增 1。当计数器的累加值达到它的预置值 60 时，计数器的完成位置 1。当 C5:0 计数器的完成位置 1 后，将使梯级 002 中的加计数器（C5:1）的值增 1，C5:1 的预置值设为 24。当 C5:1 计数达到其预置值 24 时，其完成位设置为复位。通过检查定时器的当前或累加时间值，及每个计数器的当前或累加计数值，可以显示出一天的时间。计数器 C5:1 以 24 小时的形式显示一天的小时数，而计数器 C5:0 的累加值则用来表示当前分钟数。定时器通过其当前或累加时间值来显示秒数。

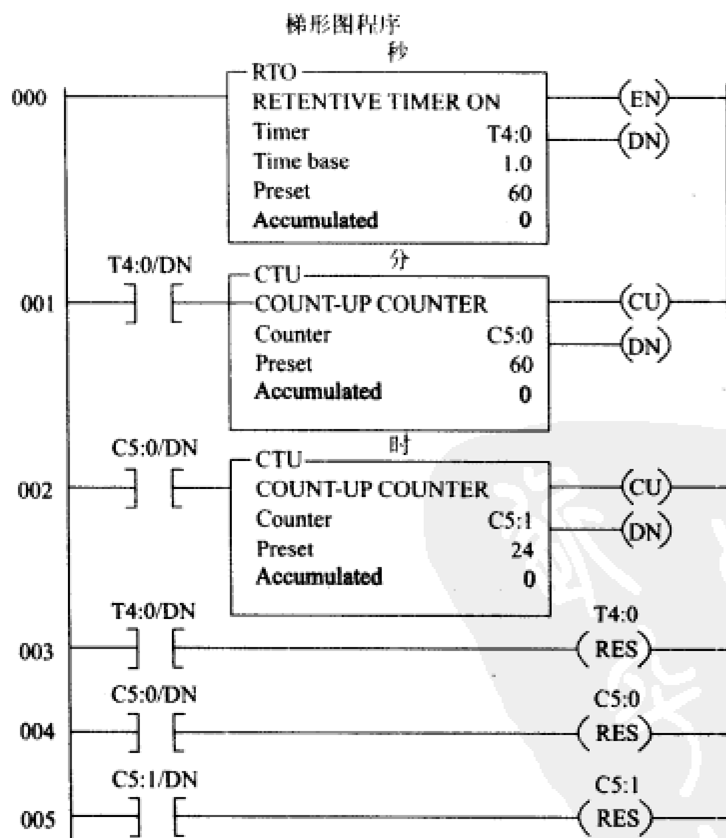


图 8-26 24 小时制时钟程序

24 小时制时钟可以用来记录一个事件的时间。图 8-27 显示了此技术的原理。在此应用中，压力开关的断开时间被记录下来。通过按下复位按钮和设置当天的时间来使电路开始工作。然后，时钟启动且使指示灯亮。无论压力开关在何时断开，时钟将自动停止并且解扣指示灯。然后，操作人员可以读出时间并且决定合上压力开关的时间。

图 8-27 事件的时间监控程序

8.5 增量式编码计数器的应用

如图 8-28 所示，随着增量式编码器轴的旋转，产生了一系列方波。随着编码器轴的旋转，当码盘使光线中断时，便会产生一个方波输出。

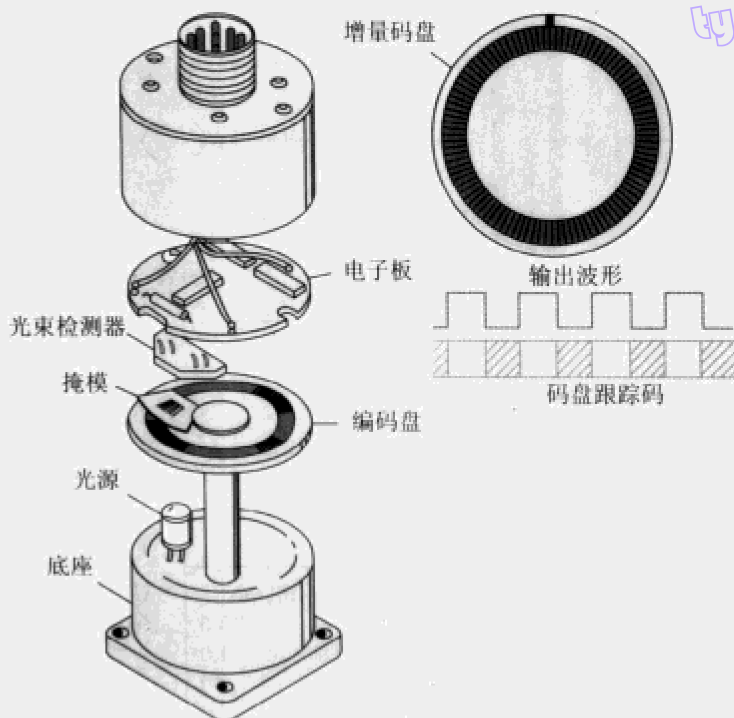


图 8-28 增量式编码器

以通过编码器输出获得的方波的个数，可以与所需要的机器的动作相对应。例如，为了将轴的旋转分为100个部分，可以选择在每旋转一周能产生100个方波的编码器。通过计数器对方波计数，可以知道轴旋转了多大角度。图8-29说明了一个将物体切割为指定尺码的例子。物体以指定的速度推进且通过编码器的脉冲进行测量，从而决定切割的正确长度。

220

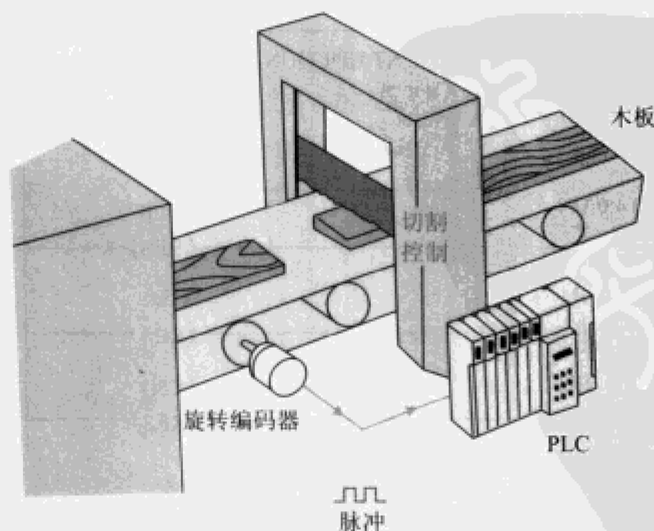


图 8-29 将切割物体为指定大小

图8-30b部分所示的程序说明了怎样用计数器用于长度的测量。系统将对通过传送机木条的总长度进行累加。程序运行可以概括如下：

- 通过磁式传感器对驱动传送机的链齿进行检测，产生输入计数脉冲。如果传送机每运动一步将有 10 个链齿通过传感器，则计数器的累加值将为步长的十倍。 221
- 通过光电式传感器对传送机上的一个参考点进行监测。当此传感器被激活，将阻止装置计数。因此，只有当木条为移动状态时，才允许计数器累加计数。
- 通过闭合复位按钮实现计数器的复位。

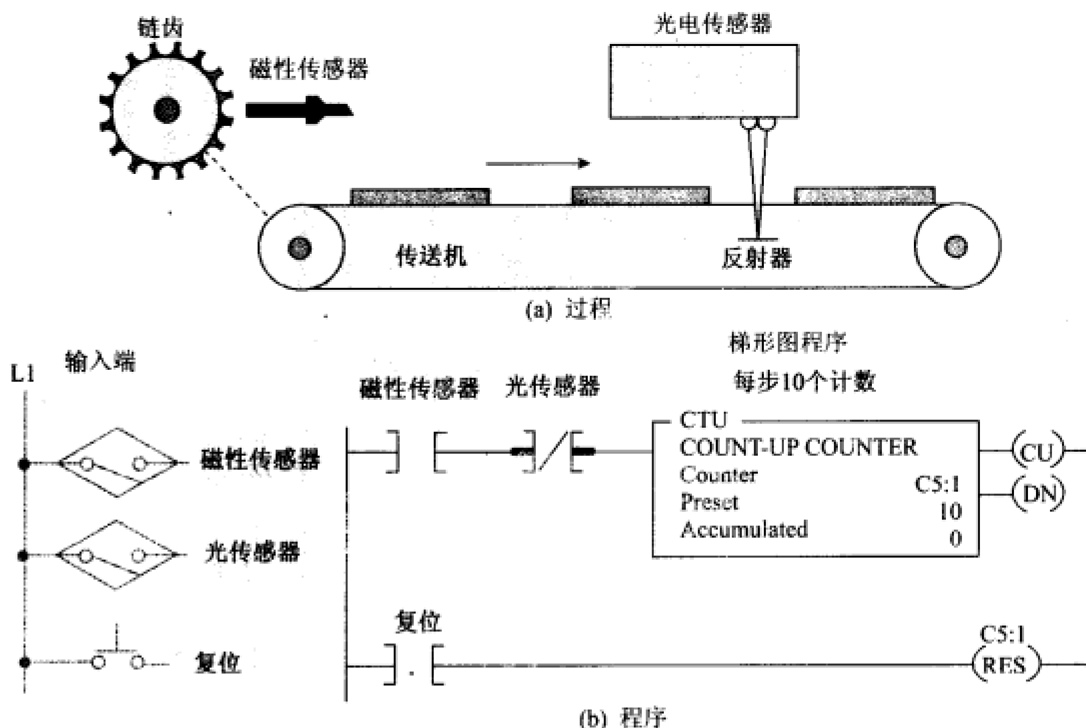


图 8-30 用计数器实现长度测量

8.6 计数器和定时器的组合

很多 PLC 的应用既用到了计数器的功能也用到定时器的功能。图 8-31 举例说明了一个自动堆物的程序，在此程序中既需要使用定时器也需要使用计数器。在处理进程中，传送机 M1 用来将金属板叠放在传送机 M2 上。每当一个金属板从 M1 掉在 M2 上，光电传感器就会给 PLC 提供一个输入脉冲。当 15 个板已经堆放完毕，通过 PLC 定时器的控制，M2 将运行 5s。程序的运行可以概括如下： 222

- 按下启动按钮，传送机 M1 开始运行。
- 当 15 个板堆放完毕，传送机 M1 停止运行，传送机 M2 开始运行。
- 传送机 M2 运行 5s 后停止，然后自动重复前过程。
- 通过定时器的完成位使定时器和计数器复位，且提供一瞬时脉冲使 M1 重新启动。

图 8-32 显示了一个电动机封锁程序。此程序的作用是使操作人员在一个小时内启动电机的次数不能超过 5 次。程序的运行可以概括如下： 223

- 当检测到过载电流，过载继电器的常开触点就闭合。
- 电机每次停机都会产生一个过载信号，电机的启动电路将被封锁 5min。
- 如果电机在一个小时内启动超过 5 次，则电机的启动电路将被封锁，直到按下复位按钮才能再次启动。

❑ 当处于封锁状态时，封锁指示灯亮。

(a) 过程

(b) 程序

图8-31　自动堆物程序

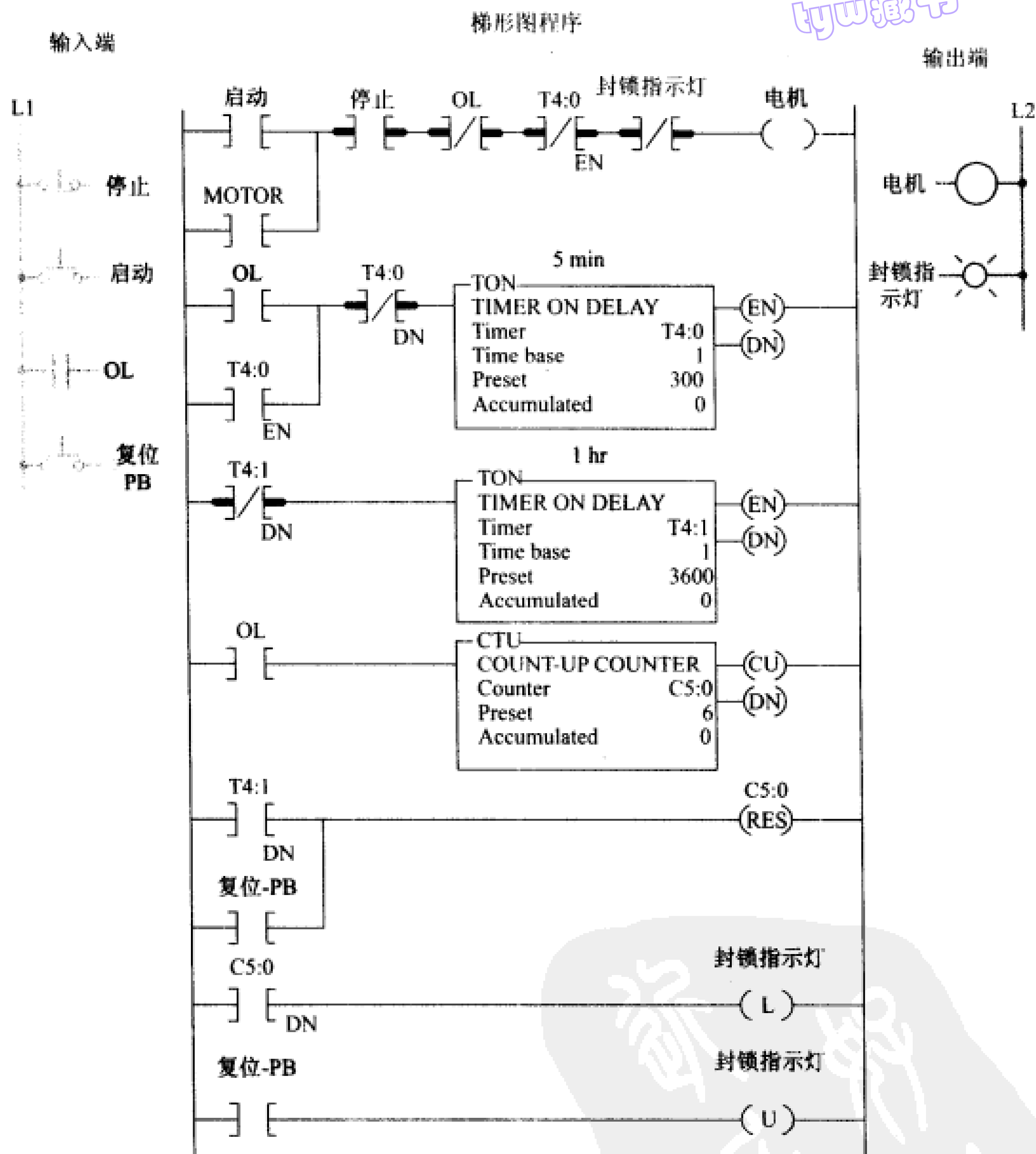


图 8-32 电动机封锁程序

图 8-33 显示了一个部件生产的流动速度显示程序。此程序的作用是显示每分钟有多少个部件通过一个给定的加工点。程序的运行概括如下： 224

- 当闭合启动开关，定时器和计数器都开始工作。
- 每当一个部件通过传感器，将会对计数器产生一个脉冲。
- 计数器计数和定时器定时将持续 1min。
- 当 1min 完成，定时器的完成位使得计数器的梯级状态为逻辑假。此时，传感器继续发出脉冲，但不会影响计数器的值。计数器的累加值代表了在先前时间通过的部件的数目。
- 通过启动开关的开闭实现工作流程的复位。



图 8-33　部件流动速度显示

[225] 当需要延时很长时间时，有时可以用定时器来驱动计数器。例如，如果需要一个定时器能定时 1 000 000s，可以使用一个定时器和计数器来达到要求。图 8-34 显示了怎样通过对定时器和计数器编程实现此目的。定时器 C4:0 的预置值为 10 000，计数器 C5:0 的预置值为 100。定时器 T4:0 的输入触点闭合 10 000s 后，它的完成位使 T4:0 复位，且计数器 C5:0 的累加值增 1。当定时器 T4:0 的完成位开关 100 次后，输出指示灯亮。因此，输出 [226] 指示灯定时器输入触点闭合经过 10 000s × 100 即 1 000 000s 后发亮。



[227]
图 8-34　通过定时器驱动计数器产生一个极大延时时间

思考题

1. 写出三种 PLC 计数器指令形式的名字，且解释各自的基本操作。
2. 列出四条与 PLC 计数器指令相关的信息。
3. 在 PLC 计数器指令中，计数器和复位指令的编址运用了什么规则？
4. PLC 计数器在什么时刻接通？
5. PLC 计数器指令的当前值在什么时候增大或减小？
6. PLC 计数器指令通常情况下都是自保持型的。解释其含义。
7. a. 比较标准 PLC 常开触点和其上升沿触点的操作。
   b. 同计数器一起使用的过渡触点通常具有什么功能？
8. 根据下列情况，确定需要选择的计数器类型：
   a. 对移动的零件总数进行计数。 b. 在过程的进出阶段保持对零件数目的计数。
   c. 在一个箱子里装有 10 个零件。随着零件的取出，对箱子里剩余的零件数进行计数。
9. 描述将两个计数器进行级联的基本编程过程。
10. 除了计数测量，还有什么类型的测量可以通过使用计数器来执行？
11. a. 加计数器的上溢位在什么时候置 1？ b. 减计数器的下溢位在什么时候置 1？
12. 描述两种常见计数器的运用。
13. PLC 计数器能够计数的最大速度由什么决定？为什么？

228

习题

1. 研究图 8-35 中的梯形图程序，回答下列问题：
   a. 编程所用的计数器是什么类型？ b. 输出端 O:2/0 在什么时候导通？
   c. 输出端 O:2/1 在什么时候导通？
   d. 假设当累加值为 24 时，控制器的交流电源消失。当电源重新恢复，累加值为多少？
   e. 梯级 4 状态为真，当梯级 1 的状态经过 5 次由假变为真的变化时，计数器的累加值为多少？
   f. 什么时候计数值将增加？ g. 什么时候计数值将被复位？

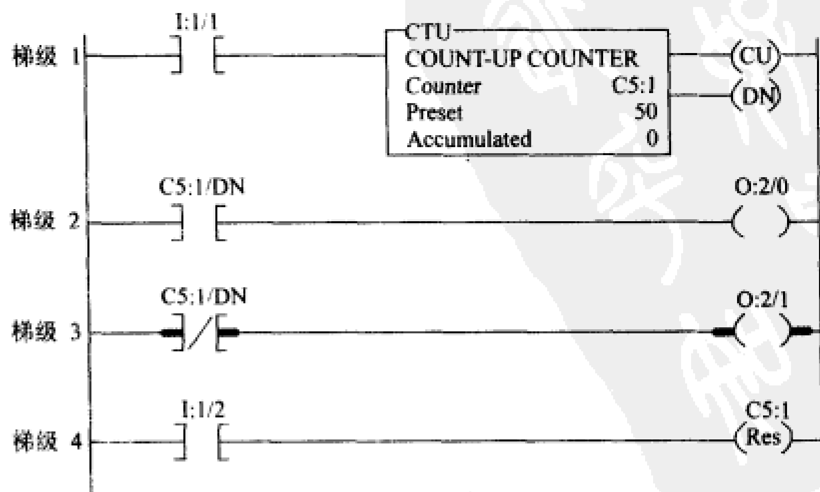


图 8-35

229

2. 研究图 8-36 的梯形图程序，回答下列问题：

a. 假设输入按钮由断开变为闭合且保持其状态。输出端 B3:0/9 的状态将受到什么影响？

b. 假设输入按钮又变为常开且保持其状态。输出端 B3:0/9 的状态将受到什么影响？

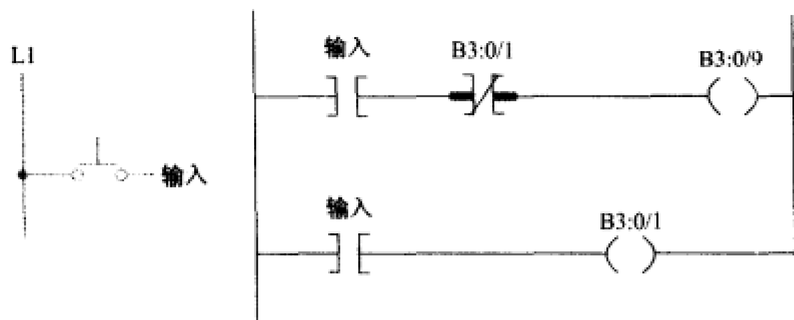


图　8-36

3. 研究图 8-37 的梯形图程序，回答下列问题：

a. 编程所用的计数器是什么类型？

b. 什么输入地址将使计数器的值增加？

c. 什么输入地址将使计数器的值减少？

d. 什么输入地址将使计数器的值清零？

e. 输出端 O:6/2 在什么时候导通？

f. 假设计数器先被复位，然后输入端 I:2/6 接收 15 次触发信号，输入端 I:3/8 接收 5 次触发信号。计数器的累加值为多少？

[230]

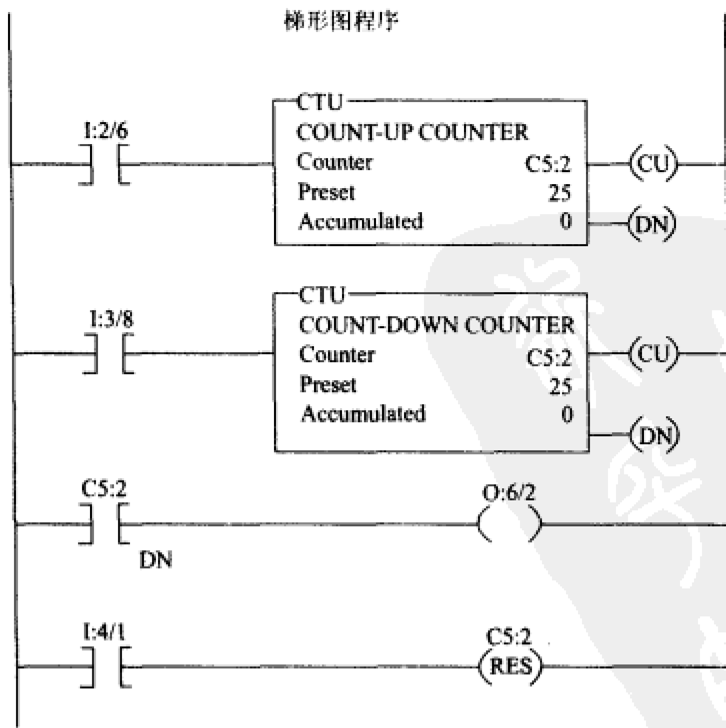


图　8-37

4. 按照以下计数器规定，设计一个 PLC 程序，画出典型 I/O 接线图和梯形图程序：

❑ 对第一个按钮的闭合次数进行计数。

- 第二个按钮每闭合一次，计数器的累加值减少1。
- 计数器的累加值不超过20的任意时刻，指示灯1亮。
- 当累加值大于或等于20时，第二个指示灯亮。
- 当选择开关闭合，计数器清零。

5. 设计一个PLC程序，且画出典型I/O接线图和梯形图程序，使之能正确执行以下控制电路：
   - 当开关闭合时，非保持型定时器启动（预置值为10s）。
   - 当定时超时时，定时器通过可编程过渡触点自动复位。
   - 对定时器定时达到10s的次数进行计数。
   - 当计数达到5时，计数器自动通过第二个可编程过渡触点复位。
   - 在计数达到5时，指示灯亮。
   - 当选择开关闭合时，指示灯灭且计数器清零。
6. 设计一个PLC程序且画出典型I/O接线图和梯形图程序使之能正确执行图8-38中的工业控制流程。操作顺序如下：
   - 产品就位（限位开关LS1触点闭合）
   - 按下启动按钮，传送电机启动，产品向位置A运动（当触动臂回到正常状态时，限位开关LS1触点断开）。
   - 产品通过传送机到达位置A后停止（位置的检测依靠编码器产生的8个上升沿脉冲，脉冲数由加计数器计数）。
   - 在经过10s的延时后，产品通过传送机移动到限位开关LS2后停止（当产品碰撞触动臂时，LS2触点闭合）。
   - 生产流程在任何时候可以通过紧急停止按钮停止。
   - 如果生产过程被紧急停机中断，计数器和定时器自动复位。

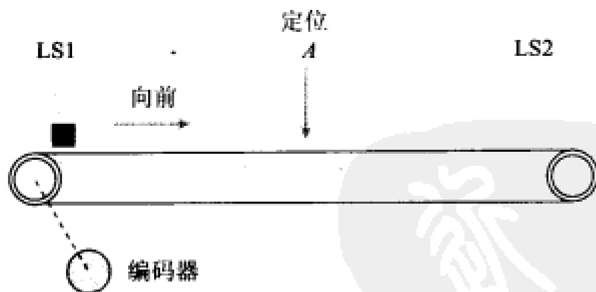


图　8-38

7. 根据图8-22所提到的加/减计数器程序回答下列问题。假设事件的发生按照以下顺序：
   - 输入端C瞬时闭合。
   - 输入端A产生20个下降沿输入信号。
   - 输入端B产生5个下降沿输入信号。

   结果：

   a. CTU计数器的累加值为多少？
   b. CTD计数器的累加值为多少？
   c. 输出端A的状态是什么？
   d. 输出端B的状态是什么？
   e. 输出端C的状态是什么？
8. 根据图8-39所示的过程编写程序。加计数器作为编程中的一部分，其作用是在质量控制中对自动分类的零件进行批量计数。为了控制质量或以检查为目的，每通过1000个零件后，将会有1个零件从另一通道分离出去。流程的操作如下：
   - 启动/停止按钮用于使传送机的电机运行或停机。
   - 接近传感器对传送机上通过的零件数进行计数。

- 当计数到 1000 时，计数器的输出信号驱动门螺线管，使零件从检查路线通过。
- 门螺线管的接通状态持续 2s，在此时间内足以使零件通过质量检查通道。
- 当 2s 区间结束，门螺线管回到其正常状态。
- 计数器清零且继续累加计数。
- 复位按钮用于对计数器进行人为复位。

232

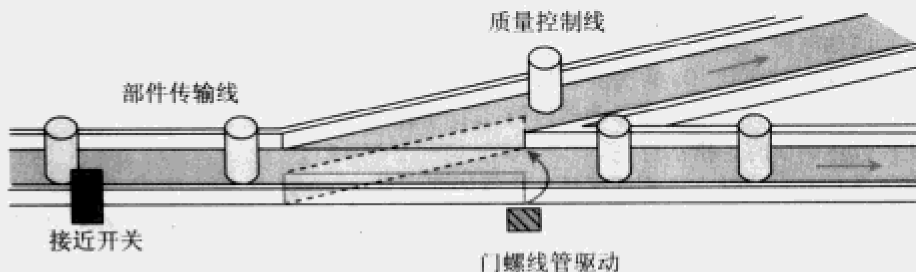


图 8-39

9. 编写程序使计数器的累加值在每经过 60s 后增加 1。在第一个计数器的累加值每次达到 60 后，第二个计数器的累加值增 1。当第一个计数器的累加值达到 60 后将复位，且第二个计数器的累加值达到 12 后将复位。
10. 根据图 8-40 所示的过程编写程序。一个电子装配套件的公司，需要使用计数器来对绕制在套件上的电阻个数进行计数。当电阻数量达到 100 时，控制器将使卷轴停止运行。然后，工作人员将切断电阻条并且对另一个套件进行绕制。操作流程如下：
    - 启动/停止按钮用于人为地使卷轴的电机启动或停止。
    - 光电传感器对通过的电阻数量进行计数。
    - 计数器的预置值为 100（每个套件上绕制的电阻数），当累加值达到 100 时，将自动停止绕制。
    - 第二个计数器用于对绕制的总数进行计数。
    - 手动复位按钮用于对每个计数器复位。

233

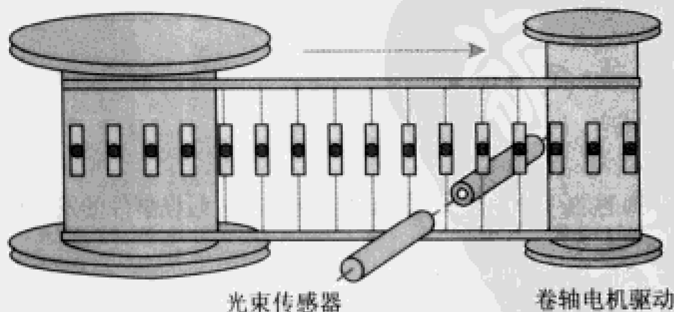


图 8-40

11. 编写程序实现如下功能：当合上输入开关后经过 20s，指示灯亮并锁存其状态。定时器将继续以 20s 的周期循环计时，直到输入开关断开后，定时器复位。在定时器第三次计时到 20s 后，指示灯灭。
12. 编写程序实现：当计数器计数到 20 时，指示灯亮。当计数到 30 时，指示灯灭。
13. 根据图 8-41 所示的箱子堆积过程进行编程。在应用中需要对传送机皮带进行控制。堆积机能够将不同数目的箱子堆积在每个模板上（取决于模板的形状和计数器的预置值）。在规定数目的箱子堆积完毕，满载模板离开且空模板置于装载位置 A 后，传送机停机。在每个

箱子通过光电传感器时，传感器将向计数器产生计数脉冲。传送机的电机除了具有启动/停止按钮，还具有远方复位按钮。当空模板置于装载位置后，操作人员能够在铲车上用远方复位按钮对系统进行复位。系统的操作可以概括如下：

- 按下启动按钮启动传送机。
- 箱子每通过光电传感器一次，计数器累加值增 1。
- 当达到预置值（此题中为 12），传送机停机。
- 铲车操作员运走满载的模板。
- 当空模板就位，铲车操作员按下远方复位按钮，整个过程重复。

234

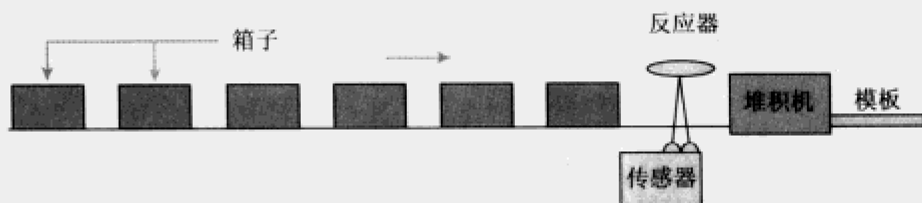


图 8-41

14. 按照以下顺序编写灯的控制程序：
   - 按下触发按钮使整个工作步骤开始。
   - 指示灯亮且保持其状态 2s。
   - 然后指示灯熄灭且保持其状态 2s。
   - 完成以上步骤后，计数器累加值增 1。
   - 然后重复以上步骤 4 次。
   - 完成第 4 次计数后，步骤停止且计数器累加值将被清零。

235



第 9 章　程序控制指令

学习目标：

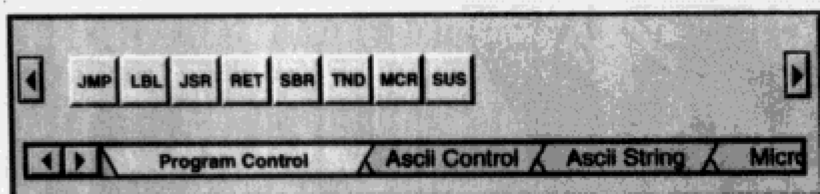
- ❑ 陈述使用程序控制指令的目的。
- ❑ 描述主控复位指令的操作，并利用它编写一个简单程序来举例说明其用途。
- ❑ 描述跳转指令和标号指令的操作。
- ❑ 解释子程序的功能。
- ❑ 描述立即输入和立即输出指令的功能。
- ❑ 描述 PLC 的强制功能。
- ❑ 描述设计一个 PLC 系统时 PLC 硬件内部以及编程的安全考虑措施。
- ❑ 描述可选定时中断和故障程序文件的功能。
- ❑ 解释暂停指令怎样用于程序的故障诊断。

本章所介绍的程序控制指令是用于改变程序的正常扫描顺序。通过使用程序控制指令，可以减少完成一个程序扫描的时间。在特定时间内没有被使用的程序可以跳过，且在程序中指定区域内的输出信号能够保持其想要的状态。本章还解释了程序控制指令在工业中的典型应用。

9.1　主控复位指令

当满足特定条件时，几种通常被称为超控指令的输出类型指令，会提供一种逻辑控制的执行部分的方法。在程序扫描中，这些程序控制指令使程序具有更大的灵活性和更高的效率。在任何特定时间都没有被使用的部分程序将被跳过，且在程序特定区域中的输出信号将保持其期望的状态。

图 9-1 显示了基于 Allen-Bradley SLC-500PLC 的典型程序控制指令，及与其相关的 RSLogix 软件。指令由超控指令组构成，超控指令组包括主控复位（MCR）和跳转（JMP）指令。通过使用有条件和无条件的分支程序和返回指令来完成这些指令。它们都运行在逻辑处理器中用户定义的区域之外。作为指令的一部分，区域的大小根据一定的方式来指定。



命　　令	名　　称	描　　述
JMP	跳转到标号	向前/后跳转到相应标号的指令
LBL	标号	指定标号单元
JSR	跳转到子程序	跳转到已设计的子程序中
RET	从子程序返回	退出当前子程序且回到先前状态
SBR	子程序	识别子程序
TND	暂停	使程序执行过程暂停
MCR	主控复位	使两条并联 MCR 指令间的输出信号清零
SUS	中止	为了程序排出故障和系统故障分析而识别特定条件

图 9-1　基于 Allen-Bradley SLC-500 PLC 的程序控制指令及与其相关的 RSLogix 软件

硬接线中的主控继电器电路用于提供整个电路的输入/输出电源的关闭。图 9-2 显示了典型的主控继电器电路。在电路中，只有在主控继电器线圈得电的情况下，才会有电流流过 MCR 触点的负载端。

237 ~ 238

主控继电器电路不能按照图 9-2 中所示的模式通过编程输入到 PLC 中，这是因为在图中包含了两个垂直的触点。因此，大多数的 PLC 制造商都将一些形式的主控继电器作为指令集的一部分。在主控继电器中，这些指令工作在相似的方式下。即当指令为真时，电路正常工作，而当指令为假时，输出端关闭。由于这些指令不是硬接线实现而是通过编程实现，因此考虑到安全因素，在提供紧急 I/O 功率关闭的主控继电器中，它们不应该作为子程序使用。

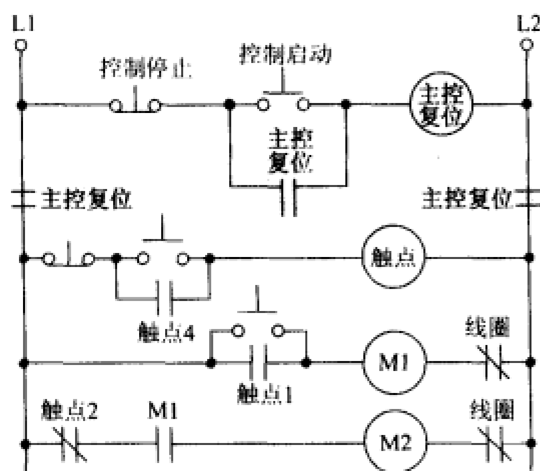


图 9-2 控制整个电路的主控继电器指令

239

通过编程，主控复位指令可以用来控制一个完整的电路或只控制电路的部分梯级。在图 9-3 的程序中，MCR 用于控制整个电路。当 MCR 指令为假或失电时，即使每个梯级的程序逻辑为真，MCR 指令下的所有非保持型（非自锁型）梯级都将失电。所有保持型梯级将保持在它们的最新状态。在用户程序中 MCR 指令确定了一个区域，在此区域中的所有非保持型输出信号都能被同时关闭。因为当保持型指令变为假时，MCR 所确定的区域能够使其维持在上一次的状态，因此在一般情况下，保持型指令不应该置于 MCR 区域中。

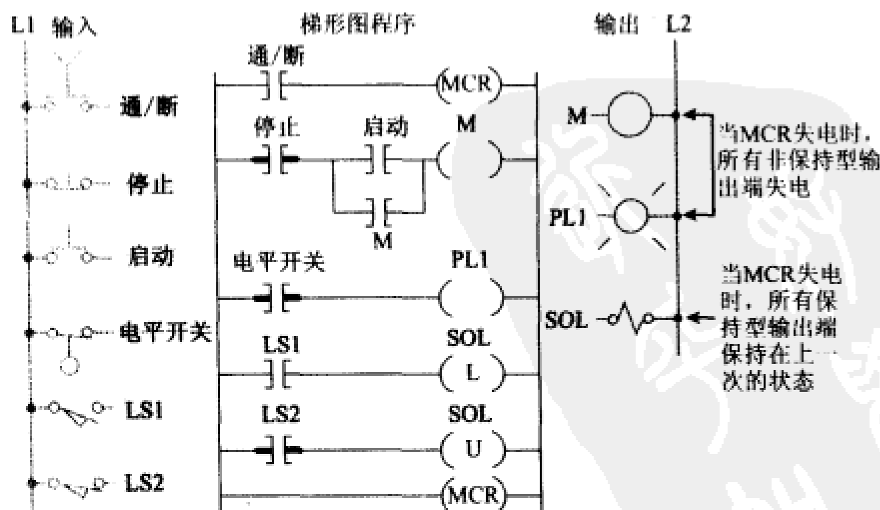


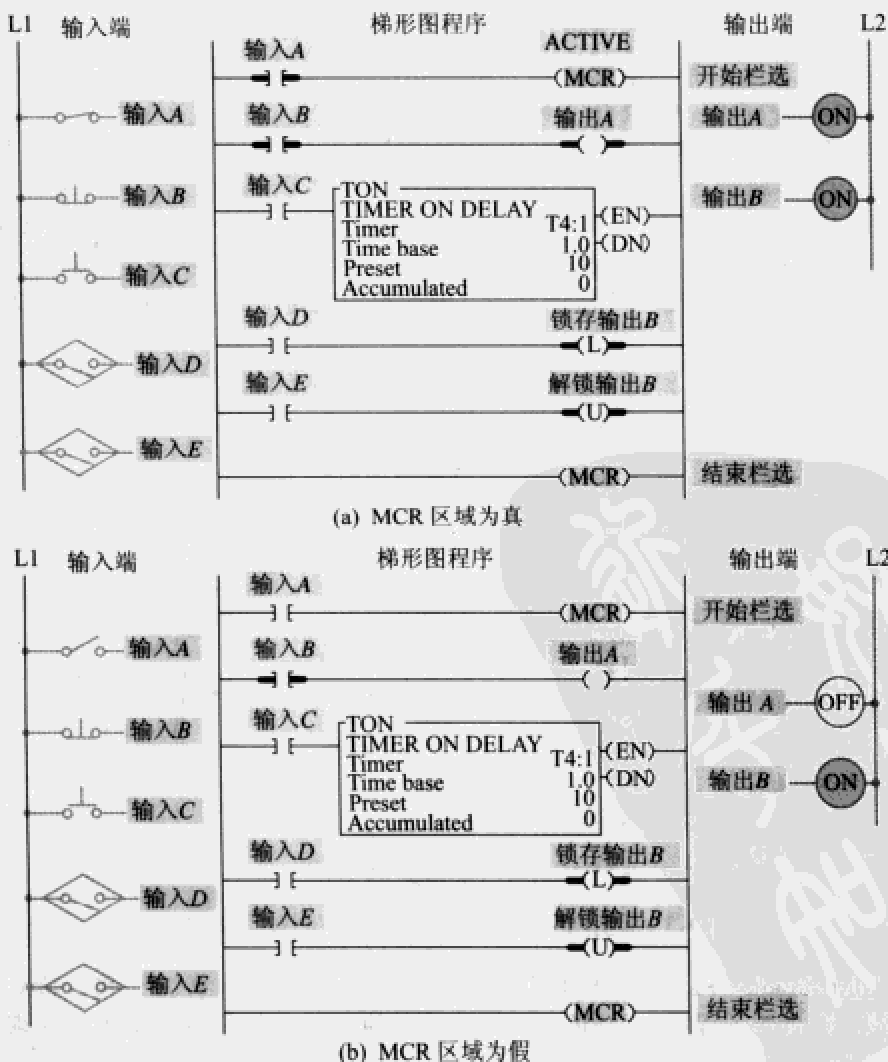
图 9-3 利用主控复位指令编程来控制整个电路

对于 Allen-Bradley PLC-5 和 SLC-500 控制器来说，程序中主控复位指令确立了一个或多个区域。在梯形图程序中，MCR 指令的成对使用可以使区域禁止操作或恢复正常操作，并且 MCR 指令没有地址。图 9-4 显示了典型的 MCR 区域编程。程序的操作可以概括如下：

- 通过区域开始和区域结束，MCR 区域被封闭。其中区域开始为一个包含带条件 MCR 的梯级，而区域结束为一个包含无条件 MCR 的梯级。
- 当启动梯级中的 MCR 为真（输入端 A 为真）时，即使区域不存在，输出端的操作仍由梯级逻辑决定。
- 当区域开始中的 MCR 为假时，区域中的所有梯级都被处理为假。程序扫描忽略了输入端，所有非保持型输出端（即输出激励指令，通延时定时器及断延时定时器）失电。所有保持型设备都维持其上一个状态，如锁存器，保持型定时器以及计数器。
- 当输入端 A 为假时，输出端 A 和 T4:1 将为假，输出端 B 将维持其上一个状态。每个梯级的输入条件将不会对输出条件产生影响。
- 在程序中 Allen-Bradley MCR 不能被嵌套，即不能被使用处于另外一个 MCR 区域中的 MCR 区域。

240

- 在一个程序中允许多个 MCR 区域的存在。



(a) MCR 区域为真

(b) MCR 区域为假

图 9-4 利用 MCR 指令编程来控制栏选区域

MCR 存储区控制指令的一种常见的应用，包括检查作为区域开始一部分的一个或多个故障位，并根据 MCR 指令中出现的故障封闭程序中不激励的部分。当检测到有故障，MCR 存储区中的输出将自动失电。

Allen-Bradley ControlLogix 控制器中的主控复位指令的操作方式同 PLC-5 和 SLC-500 控制器中的一样。当以并联的方式运用 MCR 指令时，将在两条指令之间产生一个区域，该指令没有地址。在编程时，MCR 开始的梯级具有输入指令，MCR 结束的梯级没有其他任何指令作为条件。

如果在 MCR 存储区中激活如定时器或计数器这样的指令，当区域被禁止时，指令的运行也将停止。当对包含一个 MCR 存储区的程序进行故障检测时，为了正确地修改电路，需注意到哪个梯级处于该区域中。

MCR 控制的区间必须只能包含两条 MCR 指令——一条用于定义开始且另一条用于定义结束。MCR 区域绝对不允许重叠或嵌套。任何其他的 MCR 指令或用于编程实现跳转到一个 MCR 区域的跳转指令，能对程序和机器的运行产生无法预测的和破坏性的结果。

9.2 跳转指令和子程序

同计算机编程一样，如果满足特定条件，有时可以跳过特定的程序。跳转指令（JMP）是一条用于实现这个目的的输出指令。跳转指令的好处是通过跳过执行一些指令可以减少处理器的扫描时间，被跳过执行的指令通常与当时机器的运行没有关系。跳转指令的另外一些功能如下：

241

- PLC 能保持多个程序，且只对其中对操作设备适用的程序进行扫描。
- 当出现生产故障时，部分程序可以跳过。

大多数的 PLC 制造商将跳转指令作为 PLC 指令集的一部分。一些制造商提供了跳跃指令，其本质与跳转指令相同。通过使用跳转指令，可以转移或跳转到一段程序的不同部分（如图 9-5 所示），且所有受影响的输出端都将保持其上一个状态。跳转一般允许向前和向后两个方向。跳过计数器和定时器指令时，它们的累加值不发生改变。

在 Allen-Bradley PLC 中，同时运用跳转指令（JMP）和标号指令（LBL）可以使扫描跳过一部分程序。标号为跳转提供一个目的地；它在梯级中作为第一条指令并且通常为真。调转指令使程序转移到具有相同地址的标号处。处理器跳过的程序区域由在程序中跳转指令和标号指令的位置确定。如果跳转线圈得电，在跳转指令和标号指令之间的所有逻辑将被绕过，并且处理器将在 LBL 指令之后继续扫描。

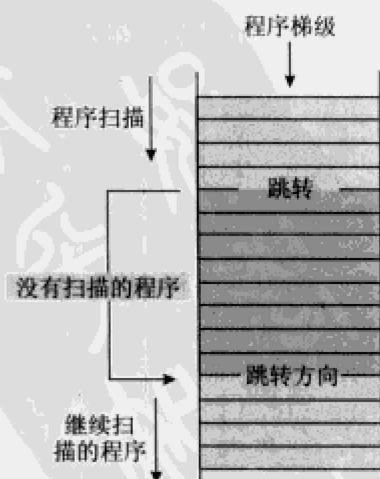


图 9-5 跳转操作

图 9-6 显示了一个跳转到标号编程的简单例子。标号指令用于识别跳转的目标梯级。标号地址的编号必须与所使用的跳转指令相匹配。标号指令对逻辑连续性没有影响，对所有实际目的，标号指令通常为逻辑真。当梯级 4 具有逻辑连续性时，程序跳转到梯级 8 继续执行。被跳过的梯级 5、6 和 7 不会被处理器扫描。输入条件没有被检测，且由这些梯级控制的输出保持上一个状态。跳转区域中的程序定时器或计数器都停止运行，且在这段时间内不会更新它们的累加值。因此应当在主程序中跳转部分之外对定时器和计数器进行编程。

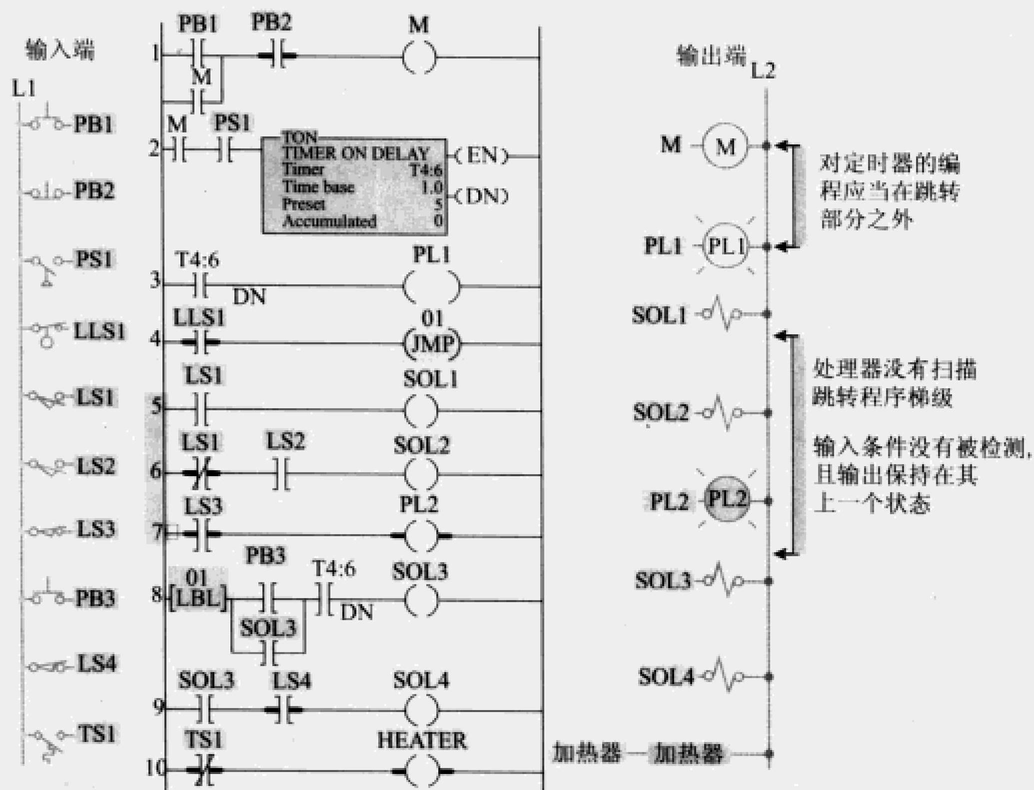


图 9-6　跳转到标号程序

如图 9-7 所示的程序，可以从多个跳转位置跳到相同的标号。在此例中，有两个编号为 20 的跳转指令，有一个编号为 20 的标号。扫描能够从任一跳转指令转到编号为 20 的标号，这取决于输入端 A 或输入端 D 是否为真。

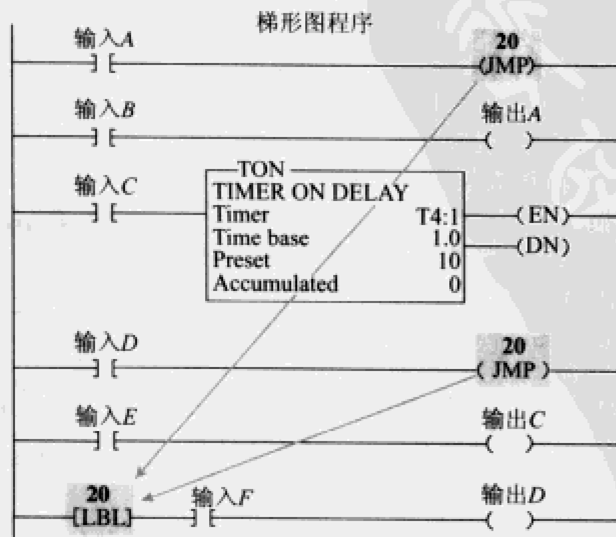


图 9-7　从两个位置跳转到标号

在程序中，也可以向后进行跳转，但是跳转的次数不应过多。必须注意到在一个循环中的扫描时间不应该太长。处理器中具有一个看门狗定时器，它设置了整个程序的最大允许扫描时间。如果超过这个时间，处理器将报错且停机。

在 SLC-500 或 PLC-5 中，JMP 的编号从 0 到 255，相应的标号具有相同的编号。在 ControlLogix 控制器中，JMP 和相应的 LBL 具有相同的名称。该名称可以多达 40 个字符，包括字母、数字和下划线。 242

在程序中绝对不允许跳转至 MCR 存储区。在 MCR 存储区中，只要 MCR 存储区为真，从 LBL 指令开始到 MCR 结束指令之间的指令程序，总是会被执行，而不需要考虑 MCR 指令起动的状态。

和计算机编程一样，在对 PLC 进行编程时，另一个很有用的方法就是先脱离主程序，执行特定功能的子程序，执行完后又返回到主程序。一个机器运行周期中，循环中的一部分需要被重复执行多次，这种情况下，采用子程序的编程方法，可以避免大量的重复编程。子程序的概念对于所有 PLC 都是一样的，但是调用子程序和从子程序返回，根据不同的 PLC 制造商，需要使用的指令也不同。对于 Allen-Bradley 控制器来说，当包含跳转到子程序（JSR）指令的梯级为真时，子程序将开始起作用。然后 CPU 将在子程序中的 LBL 处寻找目的地址（如图 9-8 所示）。通常子程序执行完后都应当返回。返回到主程序的梯级通常是没有条件的。从子程序退出后，总是返回到主程序中 JSR 指令的下一个梯级处。当包含 JSR 的梯级变为假时，在子程序中的所有输出将保持上一个状态（接通或断开）。 243

图 9-9 显示了三条与 Allen-Bradley 的子程序相关的指令，各自的功能概括如下。

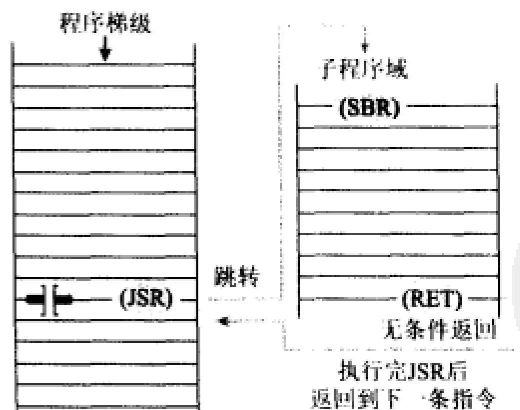


图 9-8 跳转到子程序的操作

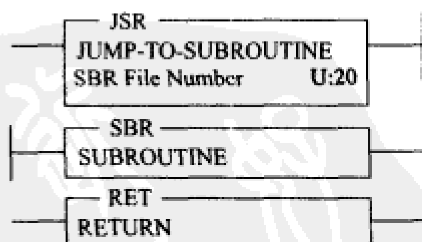


图 9-9 与子程序相关的指令

JSR。JSR 指令用于使扫描跳转到指令中指定的程序文件。输入指令的参数是唯一的。对于此输出指令来说，当梯级状态为真时，处理器在执行程序扫描时将跳转到目标子程序文件。每个子程序必须具有独立的文件编号（十进制数 3-255）。

SBR。在子程序的文件中，SBR 指令是第一个梯级中的第一条指令。SBR 是程序文件为子程序的标识符。在 JSR 指令中，文件的编号用于识别程序应当跳转到的目标。文件的编号通常为真，虽然它的使用是随意的，但通常采用推荐值。 244

RET。RET 作为一条输出指令，用于子程序文件的结束。RET 指令用于使扫描返回到 JSR 指令后的主程序。如果没有 RET 指令，扫描将从文件末返回。如果含有 RET 指令的梯级先于子程序的结束，此梯级是有条件的。在这种方式下（RET 指令出现在子程序结束之前），只要 RET 的梯级条件满足，处理器就会忽略子程序 RET 指令后面的部分。

图9-10显示了带有指示灯显示子程序的物料传送系统。如果传送机上物料的重量超过预置值，螺线管就会失电且报警指示灯开始闪烁。当称重传感开关闭合时，JSR被激活且处理器扫描跳转到子程序区域。子程序被连续扫描，并且指示灯闪烁。当传感开关断开时，处理器将不再扫描子程序区域，报警指示灯返回到接通状态。

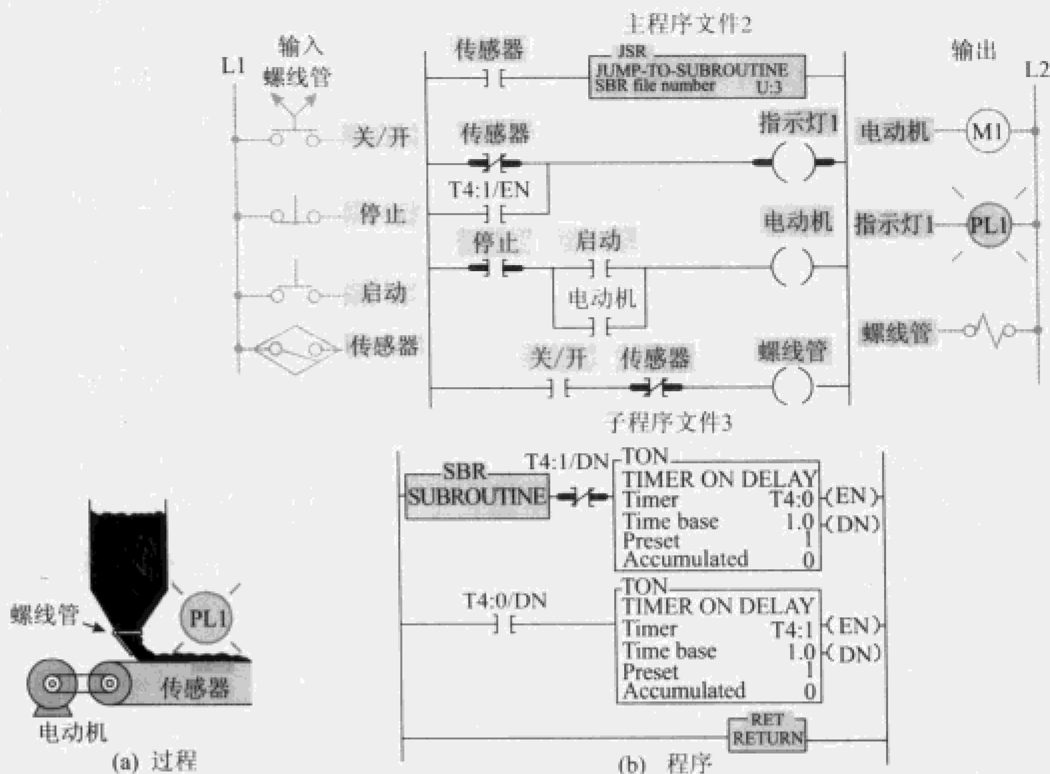


图9-10　指示灯闪烁子程序

Allen-Bradley PLC-5和SLC-500控制器的子程序与主程序存储在不同的程序文件中。主程序存储在程序文件2中，而子程序被分配在编号为3到255的程序文件中。每个子程序必须在分配给它的独立编号的程序文件中编写。图9-11显示了跳转到子程序指令、子程序和返回指令的使用。建立一个子程序的过程如下：

- ❑ 指出子程序中应当被调用的每个梯形图单元。
- ❑ 在每个单元建立一个子程序文件。每个子程序文件由SBR指令开始。
- ❑ 在每个子程序被调用的梯形图单元中，使用JSR指令来规定子程序区域的编号。
- ❑ RET指令为可选的。
  - －子程序结束后将返回到主程序。
  - －如果需要在执行到程序文件末之前就结束子程序，可以使用带条件的返回指令（RET）。

一条可选的SBR指令是用于存储输入参数的最前面的指令。这个特征使得在子程序执行前就将选择的参数值送入子程序，因此子程序能够执行数据的数学运算或逻辑运算，并且将结果返回到主程序。例如图9-12中所示的程序中，当输入端A为真时，程序的扫描将从主程序文件跳转到程序文件4。当扫描跳转到程序文件4时，数据也将从N7:30送入N7:40。当扫描从程序文件4返回到主程序后，数据将从N7:50送入N7:60。

245

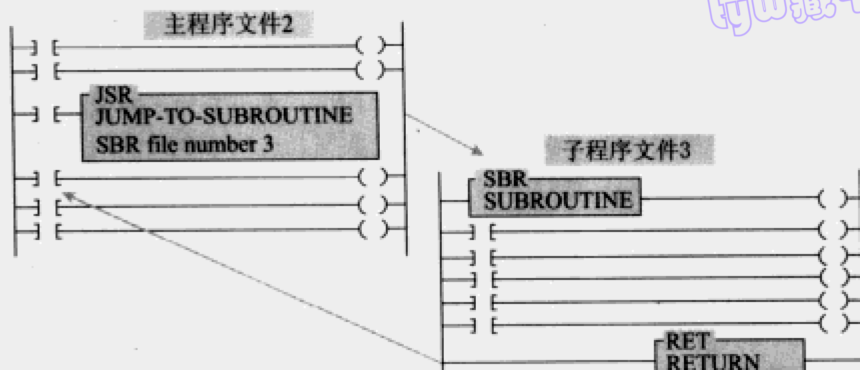


图 9-11 建立子程序文件

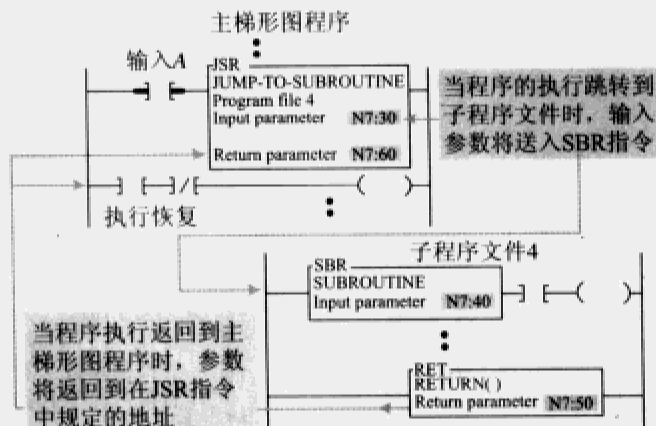


图 9-12 子程序参数的送入

如图 9-13 所示，嵌套式子程序可以使程序执行从主程序跳到一个子程序，然后又跳到另一个子程序。嵌套式子程序使得复杂的程序变得更简单且程序运行的速度更快，因为程序员不必使程序的执行连续地从一个子程序返回，然后又进入另一个子程序。由于当子程序被扫描时，主程序没有被扫描，因此嵌套式子程序的编写可能会导致扫描时间上的问题。在扫描主程序时，过长的延迟将导致输出的操作信号比要求的信号晚。通过使用立即输入/输出指令更新特定的 I/O，可以避免这种情况的发生。 246～247

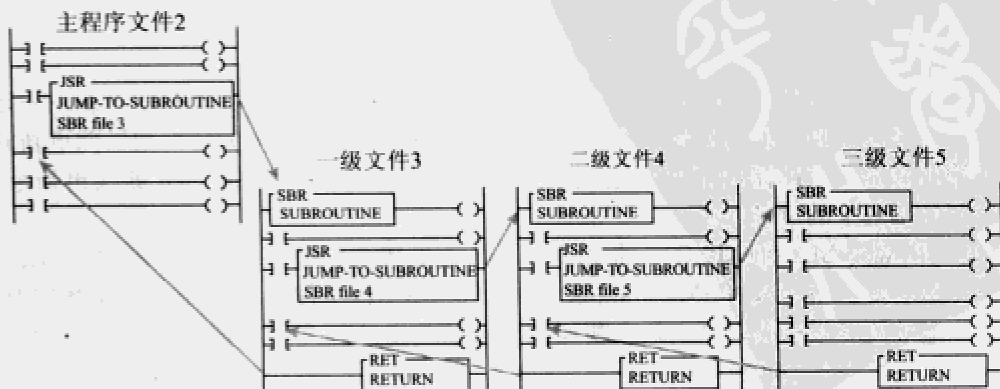


图 9-13 嵌套式子程序

9.3 立即输入和立即输出指令

立即输入和立即输出指令通过中断正常的程序扫描，来更新输入映像表文件的当前输入数据，或更新输出模块组的当前输出映像表文件数据。这些指令只能用于时间或计时处于临界值状态的区域。

Allen-Bradley PLC-5 立即输入（IIN）指令用于在执行 I/O 更新前读入输入状态。当执行此指令时，将中断程序的扫描。在执行完该指令后，恢复正常的程序扫描。此指令用于需要在扫描 I/O 端口之前，更新的设备的临界输入情况下。

立即输入指令的运用如图 9-14 所示。当程序的扫描执行到立即输入指令时，扫描中断且编址字的位被更新。如果与处于临界状态的输入设备相关的指令，正位于程序的中间或朝着程序末端执行，此时最能体现立即输入指令的用途。由于 I/O 端口的扫描在执行程序的开始阶段就已经完成，因此立即输入指令没必要设定在靠近程序开头的地方。虽然立即输入指令加快了位更新的速度，但扫描中断时间同时也增加了程序总的扫描时间。

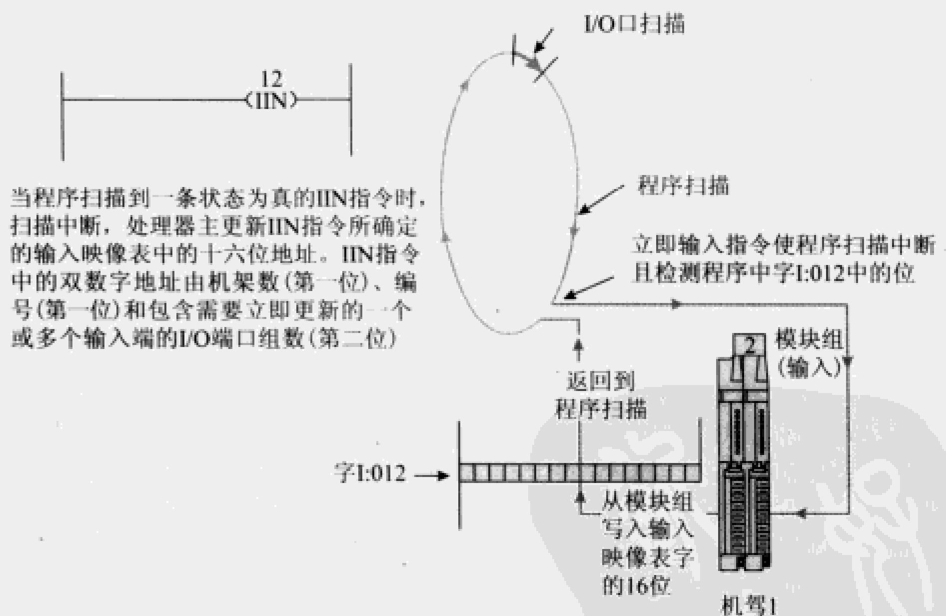


图 9-14 立即输入指令

Allen-Bradley PLC-5 立即输出（IOT）指令是输出激励指令在执行 I/O 更新前，用于更新输出设备状态的一种特殊输出激励指令。立即输出指令用于在 I/O 口扫描前更新状态的临界输出设备。立即输出指令的运用如图 9-15 所示。当程序扫描到立即输出指令时，扫描中断且编址字的位被更新。

Allen-Bradley SLC-500 PLC 中的立即 I/O 指令在 PLC-5 的基础上有所改进。SLC-500 中，被称为屏蔽立即输入（IIM）和屏蔽立即输出（IOM）的指令，允许程序员指定 16 位中的某位地址从输入模块复制到输入映像表（或从输出映像表复制到输出模块）。输入映像表或输出模块中其余的位不会被指令所影响。另外，SLC-500 指令允许从单个输入模块输入或输出一串字型数据或向一个输出模块输出一串字型数据。

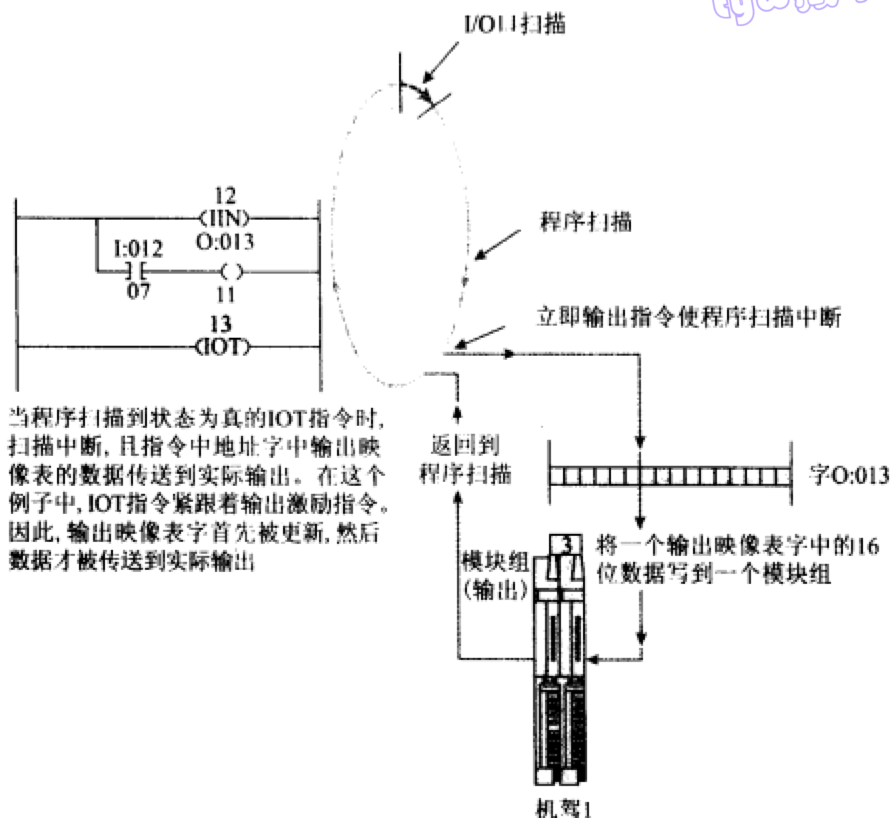


图 9-15　立即输出指令

屏蔽立即输入（IIM）指令的符号如图 9-16 所示。IIM 指令对分配了一个插槽特殊字的输入端进行操作。当 IIM 梯级为真时，程序扫描中断，并且特定输入插槽的数据通过屏蔽传送到输入数据文件。此数据在梯形图中紧跟着 IIM 指令的命令中是有效的。在指令中需要输入以下参数：

插槽　规定了包含将被更新数据的插槽和字。例如，I:3.0 表示 3 号输入插槽，字 0。在 SLC fixed 5/01 控制器中，总共可以有 8 个与插槽相关的字。在 SLC 5/02、5/03、5/04 和 5/05 控制器，有 32 个与插槽相关的字。

249

屏蔽　规定了一个十六进制常数或寄存器地址。对于屏蔽来说，位地址中的 1 允许数据从源地址传送到目的地址。位地址中的 0 禁止或阻碍数据从源地址传送到目的地址。

长度　用于传送每个插槽中多于一个字的情况；只存在于 SLC 5/03、5/04 和 5/05 的处理器中。

屏蔽立即输出（IOM）指令的符号如图 9-17 所示。IOM 指令对插槽中特定字地址的物理输出进行操作。当 IOM 梯级为真时，程序扫描中断，指令规定的插槽位置的模块的输出数据被更新。此数据对梯形图中紧跟着的 IOM 指令的命令是有效的。输入参数基本上与 IIM 指令的输入参数相同。

处理器与本地机架的通信速度比与远方机架的通信速度快很多倍。这是因为本地的 I/O扫描与程序扫描同步且与处理器的通信是并行的，而远方的 I/O 扫描与程序扫描异步且与远方 I/O 的通信是串行的。因此，高速运行的设备需要安装在本地机架上。

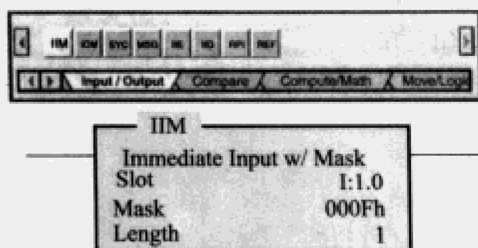


图 9-16　IIM 指令。在此例中，IIM 指令从I:1.0起检索数据并且使数据通过屏蔽。屏蔽只允许四位最低有效位被转移到输入寄存器 I:1.0，在此输入寄存器中，程序员只允许对部分输入程序进行更新

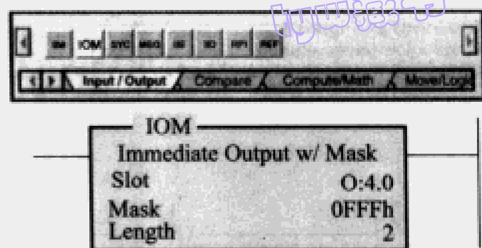


图 9-17　IOM 指令

9.4　强制外部 I/O 选址

PLC 的强制能力允许程序员通过键盘操作来打开或关闭外部输入或输出。完成此操作不需考虑输入端的现场设备（限位开关等）或输出端的逻辑梯级的实际状态。在现场故障设备能够被修理好之前，强制能力允许机器或进程能连续运行。在机器或过程的启动到检修期间，对还没有执行的程序的动作进行模拟是非常重要的。

强制输入对输入映像表文件进行位操作，因此影响了所有使用到这些位的程序区域。强制输入的执行是在输入端扫描之后。当强制操作输入地址时，是将位于 I/O 地址中指令的位强制变为导通状态或关闭状态。图 9-18 显示了一个输入怎样被强制变为导通状态。处理器忽略了限位开关的实际状态（断开），而将输入端 I:1/3 当作导通状态。程序扫描对此进行了记录，且程序在此强制状态下执行。换言之，程序在假设限位开关是闭合的情况下执行。

250
~
251

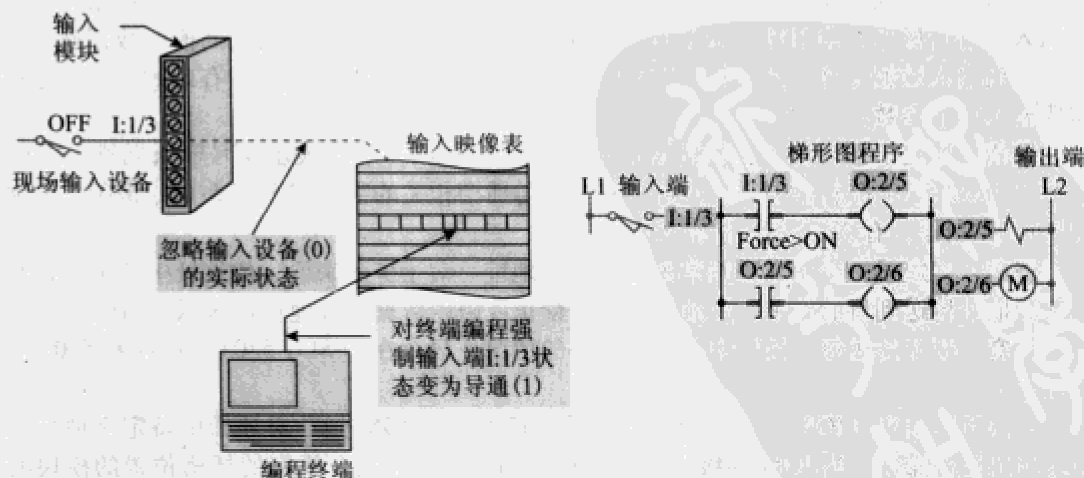


图 9-18　将输入地址强制变为导通状态

强制输出信号只影响到被编址的输出端。因此，当输出映像表文件位没有被影响时，程序就不会被影响。强制输出的执行刚好处于输出映像表文件被更新前。当对输出地址进行强制时，只将输出终端状态强制变为导通或关断。该地址输出指令的状态位通常不会被影响。图 9-19 显示了怎样对输出端进行强制接通。编程器通过与处理器连接将

输出地址 O:2/5 状态变为导通，即使输出映像表文件显示用户逻辑已经将此地址设置为断开状态。由于输出地址 O:2/5 的状态位没有受影响，因此输出地址 O:2/6 保持断开状态。

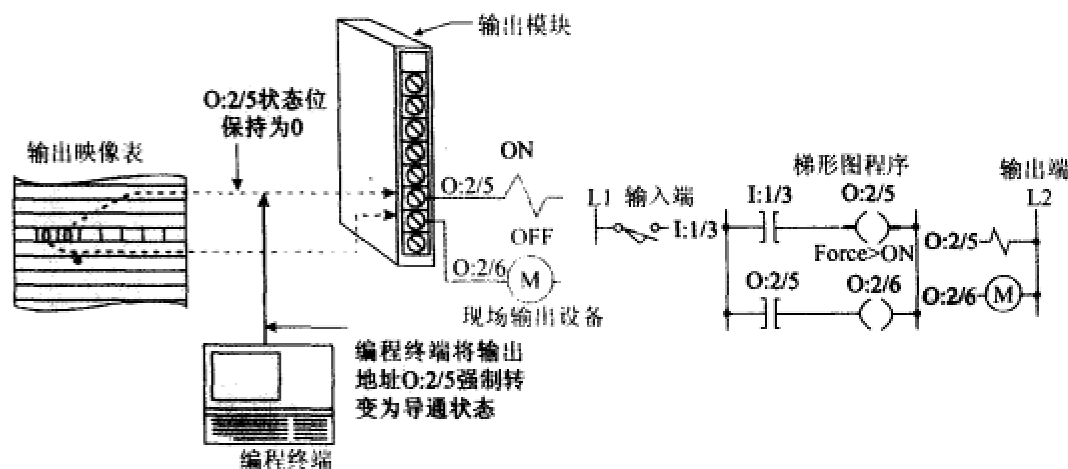


图 9-19 强制输出地址为导通状态

通过安装电路跨接线，可以实现将传统继电器控制系统的物理输入地址进行替换。在 PLC 控制中，电路跨接线是没有必要的，因为输入地址数据表中的数据状态可以强制将其转变为导通和关断。通过将外部数据位强制变为导通或关断状态，强制功能允许人为地覆盖外部输入电路的真实状态。类似地，通过将输出地址位强制变为导通或关断状态，可以人为地覆盖处理器的逻辑以及输出数据存储区位的状态。通过将输出强制变为关断，阻止控制器激励输出，即使处于正常控制下的梯形图逻辑状态为真。在其他例子中，输出地址可以强制变为导通状态，即使控制输出地址的梯级逻辑状态为假。 252

当离线状态下监视存储区或在处理器模式下在线监视存储区时，都可以输入程序且实现强制或非强制功能。利用 RSLogic 软件的执行步骤如下：

（1）在需要强制改变逻辑状态为导通或关断时，打开程序文件。

（2）用鼠标右键单击需要强制操作的 I/O 位。

（3）在主菜单中选择 Goto Data Table（转向数据表）。

（4）在显示的相关数据表中，单击 Forces（强制）按钮。

（5）数据表中被选择的位变亮。用鼠标右键单击此位。

（6）从弹出的主菜单中，可以强制将被选择位改变为导通或关断状态。

图 9-20 显示了将数据表中位 I:1/3 强制转为导通状态的强制方案。当使用强制功能时，必须谨慎。只有在用户完全了解电路和机器加工流程或驱动设备的情况下，才能使用强制功能。在操作机器时，用户必须了解强制功能对输入端或输出端所带来的潜在影响，以避免出现人员受伤和设备损坏。在使用强制功能前，应检查是否只对 I/O 点进行强制操作，或是否对用户设定的逻辑如 I/O 点有影响。当强制操作起作用时，大多数编程终端都提供有可视的报警方法。 253

在涉及旋转设备的情况下，运用强制指令会变得尤其危险。例如，当维护人员在对失电的电机进行维护时，如果其他人应用强制功能就会使电机突然带电。这就是 I/O 机架需要硬接线主控电路的原因。硬接线电路提供一种将电源从 I/O 系统完全移去的方法，因此当主控关闭时，可以确保所有输入端和输出端都不带电。

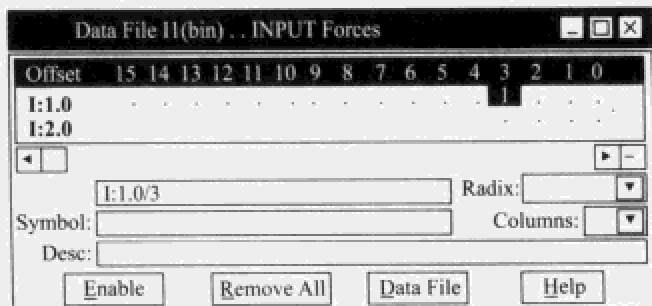


图 9-20 强制将数据表位 I:1/3 转变为导通状态的强制方案

9.5 保安电路

要想停止部分或整个控制器的运行，或者被控制的机器或过程，需要提供充分的紧急电路。这些电路应当在控制器的外部进行硬接线，这样可以在整个控制器失效时，能够独立且快速的关闭系统。

图 9-21 显示了关于 PLC 安装的典型安全接线图。主隔离开关被安装在动力线的进线端，可以实现与 PLC 系统完全电隔离。

主要电隔离开关应当安置在操作人员和维护人员能够方便快速到达的位置。理想情况是，隔离开关被装配在 PLC 外壳的外部，以至于能够不需要打开 PLC 外壳就能对隔离开关进行操作。除了要实现电隔离，还需要进行失电自锁，且在控制器对机器或流程进行控制之前，应当对所有其他电源（气动和液动）进行标识。隔离变压器用于实现将控制器与主电源分布系统隔离，且将电压降至交流 120V。

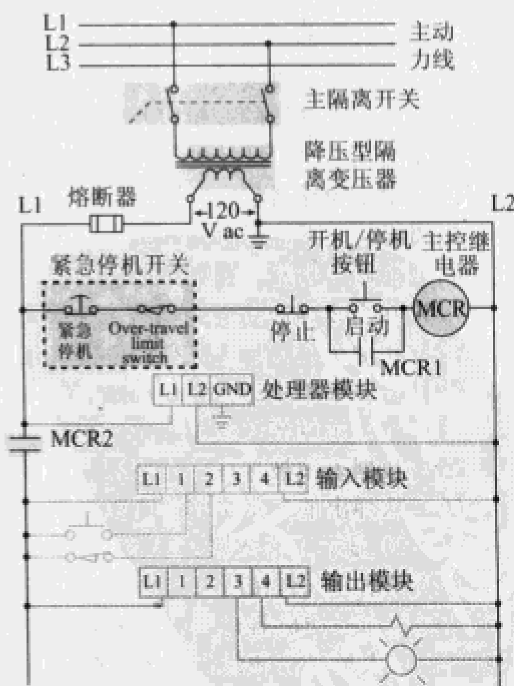


图 9-21 典型 PLC 安全接线图

采用了硬接线的主控继电器，提供了紧急关闭控制器的方便方法。从安全性考虑，由于主控继电器允许在不同位置安置紧急停机开关，因此这种安装是很重要的。行程开关或蘑菇头按钮需要串联，以使当它们其中一个断开时，主控电路失电。这样就把输入设备和输出设备电路的电源切除。而电源继续向控制器供电，这样可以使处理器模块上面的故障指示灯仍然可以被监视到。注意到主控继电器并不是隔离开关的替代物。当需要更换某个模块，或者更换输出熔断器或者对设备进行操作时，主隔离开关都应当拉闸且锁住。

254

当继电器失电，主控继电器必须能切除机器 I/O 设备的电源，禁止所有的机器运行。任何部件都可能发生故障，主控继电器电路中的开关也不例外。这些开关中的某个开关故障通常会导致电路开路，这将是个安全的断电故障。然而，如果这些开关当中的某个开关发生了短路，将不再提供任何安全保护。这些开关应当进行周期性的检查，以确保它们在

255

需要时能够使机器停止运行。切记不要改变电路而取消其功能，这可能会导致严重的人员伤亡或机器损坏。

在 PLC 编程时应当考虑到一定的安全因素。PLC 程序的可靠性是随着程序员投入的时间和思考而不断提高的。例如考虑到用电机启动器的自保持触点（如图 9-22 所示）来代替参考输出线圈指令的可编程触点。在程序中，使用外场感应启动机辅助触点的情况就现场布线和硬件方面来说花费相当大，但是由于它向处理器提供了电机精确状态的正反馈，因此使用它更安全。例如，假设在过载情况下启动器的 OL 触点断开。由于启动线圈的电能消失，电机停止运行。如果程序中用 NO（常开）触点控制输出线圈并自保持，处理器将不会检查到电机已经处于失电状态。当 OL 触点复位时，电机也将立即重新启动，产生潜在的不安全运行状态。

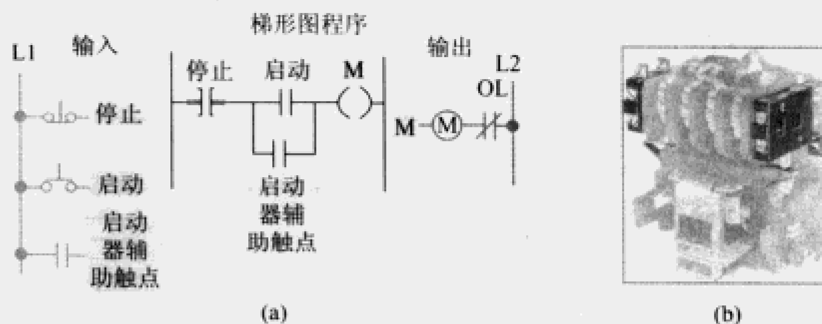


图 9-22 使用辅助触点的电机启动程序

另一个安全性考虑就是停止按钮的接线。停止按钮通常被认为具有保安功能和操作功能。正因为如此，在接线时应当使用 NC 触点，且通过编程来检验它的导通状态。在编程中使用 NO 触点来确定它的状态为断开（如图 9-23 所示），这将产生相同的逻辑关系但并不安全。假设使用后一种配置。如果由于发生一连串事件，导致位于按钮和输入点之间的电路被损坏，停止按钮将被长时间按下，但是由于输入不可能为真，PLC 逻辑不会激活停止指令。如果停止按钮控制电路的电能丢失，逻辑关系仍保持为真。如果使用 NC 触点的接线配置，输入点连续通电，只有需要停止时才断电。任何出现在停止电路中的接线故障或电路失电，都等效为一种内部的停止功能。

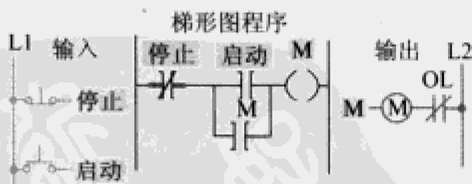


图 9-23 常开按钮停止电路

256

9.6 可选定时中断

可选定时中断（STI）指令允许中断对主程序文件自动地扫描，而在一个时基内对特定的子程序文件进行扫描。对于 Allen-Bradley SLC-500 控制器来说，被执行程序文件的时基和被指定为可选定时中断文件的程序文件，由存储在数据文件中字 30 和字 31 的状态决定。时基的长度可以从 1 ms 到 32 767 ms，存储在字 30 中值的时基为 1 ms 间隔。字 31 中存储了被指定为可选定时中断文件的程序文件，这个程序文件可以为 3 到 999 中的任意一个。在时基字中输入 0 可以使可选定时中断无效。

当部分程序需要在一个时基内而不是在一个事件基础上执行时，可选定时中断程序就被执行。例如，一段程序在重复的时间内为了得到准确结果可能需要进行一定的计算。这

些计算可以通过在可选定时中断文件中编制程序来完成。这条指令同样可以用于需要周期性注油过程的应用中。

立即输入指令和立即输出指令通常设置在可选定时中断文件，以使部分程序在一个时基内被更新。这种方法可以应用在高速生产线上，传输线上的部件通过传感器被检查时，其速度大于程序的扫描周期。通过此方法，在一个程序扫描时间内，可以对产品进行多次扫描，且在扫描结束前可以采取适当的措施。

可选定时终端禁止（STD）指令通常与可选定时中断开启（STE）指令并联来创建一个区域，在此区域中 STI 中断不会出现。图 9-24 说明了 STD 指令和 STE 指令的使用。在此程序中，假设 STI 指令已经发生作用。在梯形图程序中，梯级 6 中的 STD 指令和梯级 12 中的 STE 指令是为了避免在梯级 7 和梯级 11 之间执行 STI 子程序。STD 指令（梯级 6）对 STI 使能位进行复位，而 STE 指令（梯级 12）使使能位从新置位。梯级 0 中的第一个传送位 S：1/15 和 STE 指令是为了确保 STI 功能在完成一个周期后被初始化。

257

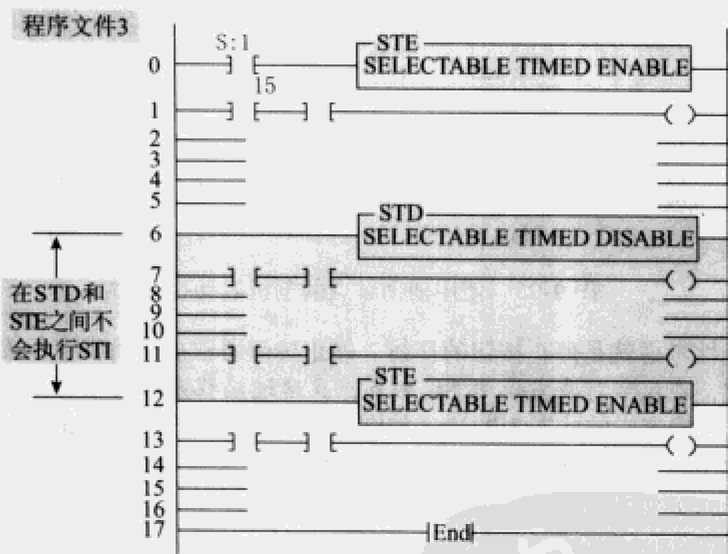


图 9-24 可选定时终端禁止（STD）指令和可选定时中断开启（STE）指令

9.7 故障程序

Allen-Bradley PLC-5 和 SLC-500 控制器允许将一个子程序文件指定为一个故障程序。如果使用故障程序，将决定处理器怎样对程序故障做出反应。被指定为故障程序的程序文件由存储在状态文件中字 29 中的值决定。在字 29 中输入 0 可以使故障程序无效。

有两种主要的故障可以造成处理器的故障：可恢复的故障和不可恢复的故障。对 PLC-5 来说，状态文件中的字 11 作为主故障字，它中的 00 位到 07 位表示可恢复的故障；从 08 到 15 位表示不可恢复的故障。当处理器检测到主要故障时，处理器将寻找故障程序。如果故障程序存在，处理器将执行此程序；如果故障程序不存在，处理器将停机。当存在一个故障程序且此故障程序为可恢复时，将执行此程序。如果此故障为不可恢复时，一旦扫描到故障程序就立即停机，或者在关闭故障程序时允许按照顺序停机。

258

9.8 暂停指令

暂停（TND）指令作为一条输出指令，用于逐步调试程序或有条件地忽略当前程序文

件或子程序中 TND 指令后的部分程序。当梯级条件为真时，此指令将停止程序扫描，更新 I/O 口状态并且在主程序区域的梯级 0 中恢复扫描。

图 9-25 显示了 TND 指令在程序检测中的作用。TND 指令使程序只在它以上部分运行。调试一段新的程序时，可以在程序中逐渐的向下移动 TND 指令。用 TDN 指令编程时可以是无条件的，或者根据调试的需要在梯级中加入条件。

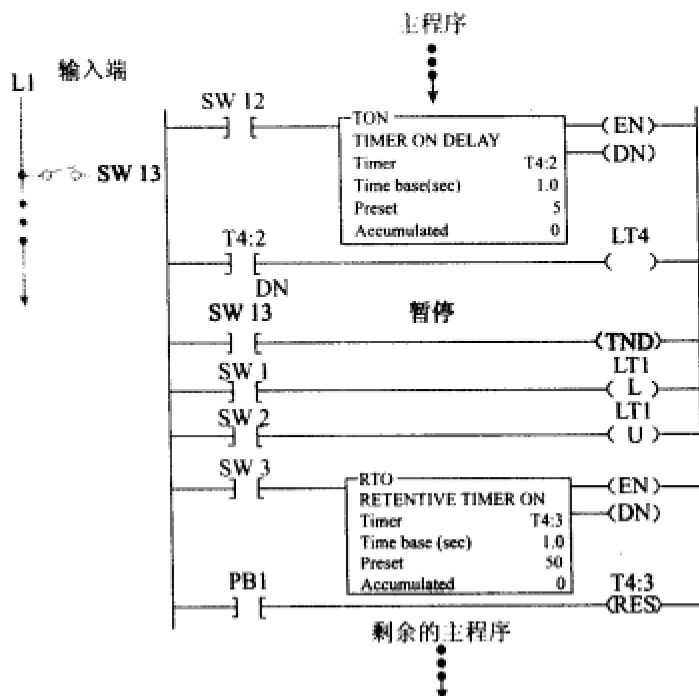


图 9-25 暂停指令

259

思考题

1. a. 在编程中用两条 MCR 输出指令来控制部分程序。阐明编程的过程。
   b. 陈述当 MCR 指令状态由假变为真时，在 MCR 指令区域内的输出设备的状态会受什么影响?
   c. 陈述当 MCR 指令状态由真变为假时，在 MCR 指令区域内的输出设备的状态会受什么影响?
2. 跳转指令最主要的优点是什么?
3. 在被跳过部分的程序里面通常不包含什么类型的指令? 为什么?
4. a. 在跳转到标号指令对中，使用标号指令的目的是什么?
   b. 当执行跳转到标号指令时，以什么方式影响到被跳过的梯级?
5. a. 阐述跳转到子程序指令的功能。
   b. 在什么类型的机器运行中，跳转到子程序指令能够节省大量重复的程序。
6. 子程序的嵌套具有什么优点?
7. a. 在什么时候使用立即输入指令和立即输出指令?
   b. 为什么在靠近程序的起始部分立即输入指令或立即输出指令对程序几乎不起作用?

260

8. a. PLC 的强制能力可以允许用户完成什么操作?
   b. 概述强制功能的两个实际用途。
   c. 当利用强制功能时，为什么需要格外小心?
9. 为什么紧急停止电路要实现硬接线而不通过编程实现?

10. 陈述以下关于 PLC 安装的基本保安接线的功能：
    a. 主隔离开关 b. 隔离变压器 c. 紧急停止 d. 主控继电器
11. 当对电机启动电路编程时，为什么用自保持辅助触点来代替参照输出线圈指令的可编程触点会更安全？
12. 当对停止按钮编程时，为什么使用 NC 按钮来检查导通条件比用 NO 按钮来检查关断条件要安全？
13. 阐述可选定时中断指令。
14. 阐述故障程序文件的功能。

261

15. 怎样用暂停指令来对程序进行故障分析？

习题

1. 针对图 9-26 所示的 MCR 程序，按顺序回答问题，假设程序已经输入到 PLC 中且 PLC 已经处于运行模式，所有开关都断开。
    a. 合上开关 S2 和 S3。输出端 PL1 和 PL2 将出现什么情况？为什么？
    b. 开关 S2 和 S3 继续保持闭合状态，合上开关 S1。是否会出现输出端 PL1 和 PL2 其中之一亮，或两者都亮的情况？为什么？
    c. 开关 S2 和 S3 继续保持闭合状态，断开开关 S1。输出端 PL1 和 PL2 都会失电吗？为什么？
    d. 断开所有开关，然后只合上开关 S6。定时器会计时吗？为什么？
    e. 开关 S6 继续保持闭合状态，合上开关 S5。定时器会计时吗？为什么？
    f. 开关 S6 继续保持闭合状态，断开开关 S5。定时器将会发生什么情况？如果定时器为 RTO 型而不是 TON 型，累加值将会发生什么情况？

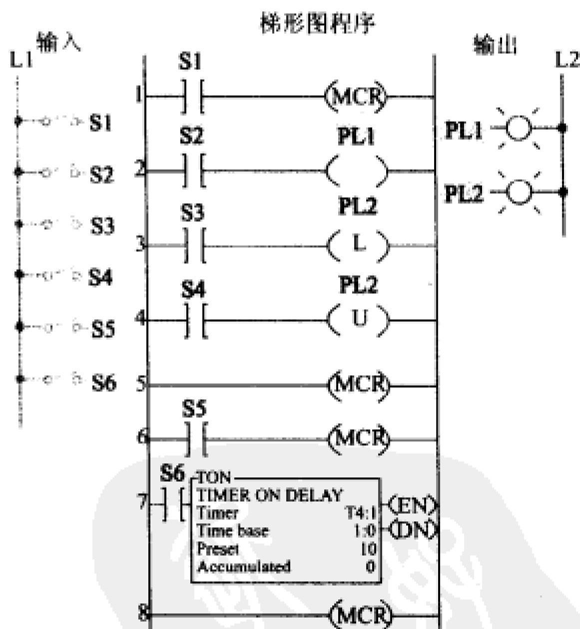


图 9-26

262

2. 针对图 9-27 所示跳转到标号程序，按照顺序回答问题。假设在完成每步操作后所有开关都处于断开状态。
    a. 合上开关 S3。输出端 PL1 会得电吗？为什么？
    b. 先合上开关 S2，然后合上开关 S5。输出端 PL4 会得电吗？为什么？
    c. 合上开关 S3 且输出端 PL1 得电。然后合上开关 S2。输出端 PL1 在 S2 合上后是得电还是失电？为什么？

263

    d. 按照以下顺序依次合上所有开关：S1、S2、S3、S5、S4。哪个指示灯将会发亮？
3. 针对图 9-28 所示跳转到子程序且返回程序，按照顺序回答问题。假设在完成每步操作后所有开关都处于断开状态。
    a. 合上开关 S1、S3、S4 和 S5。哪个指示灯不会亮？为什么？
    b. 合上开关 S2 后，然后合上开关 S4。输出端 PL3 将会得电吗？为什么？
    c. RET 指令将使程序扫描返回到哪个梯级？

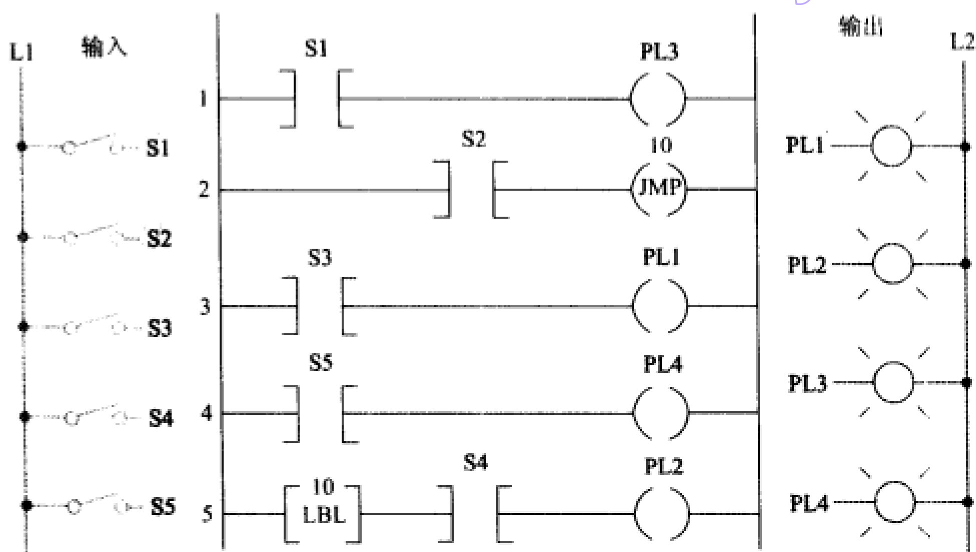


图　9-27

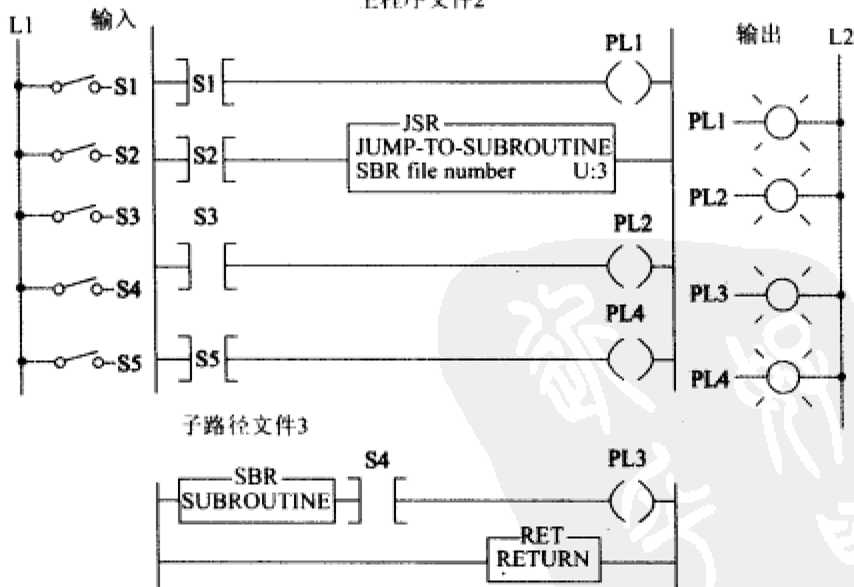


图　9-28

4. 根据图 9-29，按顺序回答问题。假设在完成每步操作后所有开关都处于断开状态。

　a. 按顺序合上开关 S2、S12 和 S5，输出端 PL5 会得电吗？为什么？

　b. 除了开关 S7 合上，其余所有开关都断开，RTO 会开始计时吗？为什么？

　c. 按顺序合上开关 S3 和 S8，指示灯 PL2 会发生什么情况？为什么？

　d. 在什么时候 TON 定时器开始工作？

　e. 假设所有开关都合上，梯级将以什么顺序被扫描？

　f. 假设所有开关都断开，梯级将以什么顺序被扫描？

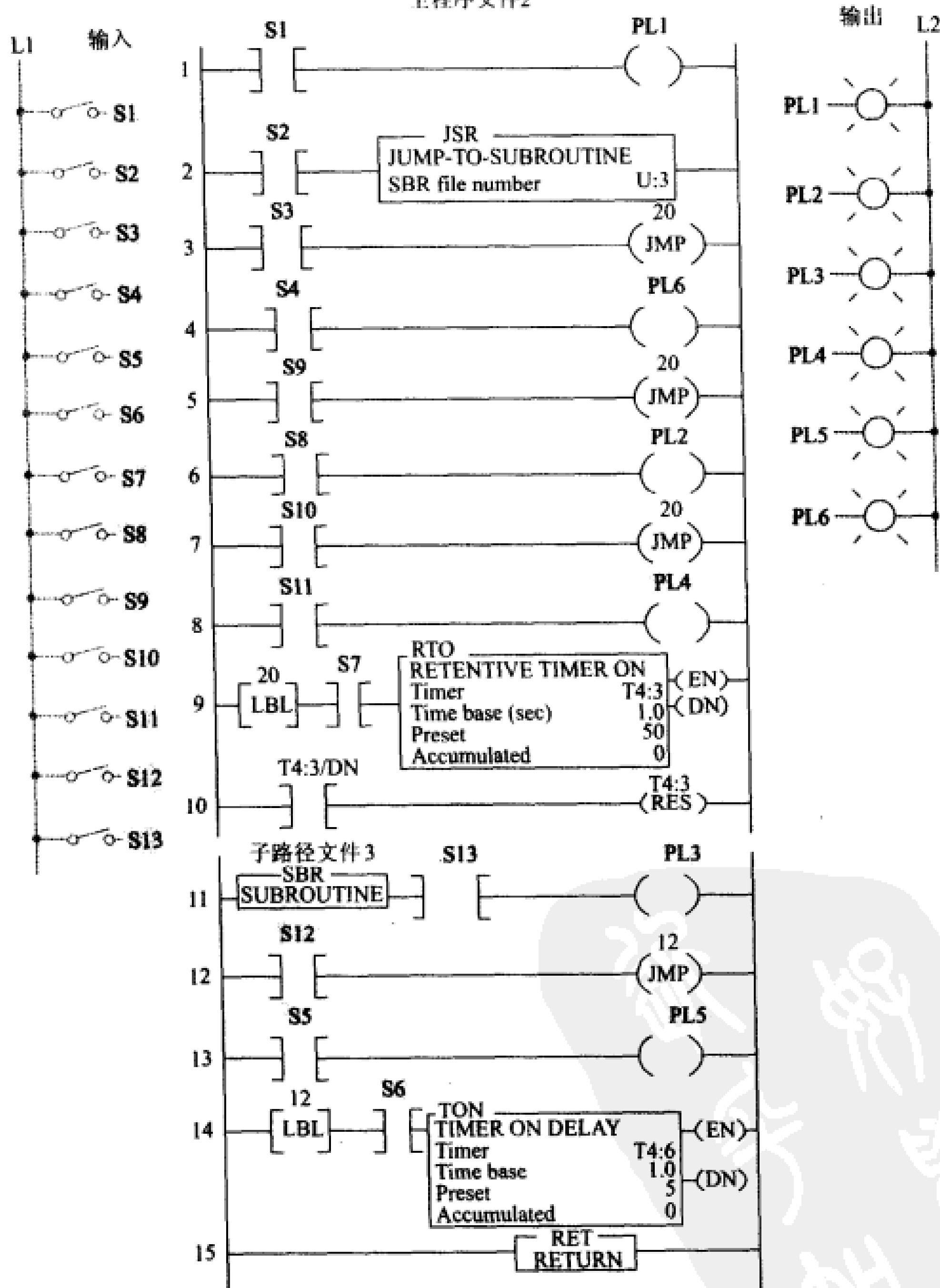

梯形图程序
主程序文件2
L1
输入
S1
S2
S3
S4
S5
S6
S7
S8
S9
S10
S11
S12
S13
PL1
JSR
JUMP-TO-SUBROUTINE
SBR file number U:3
20
JMP
PL6
PL2
PL4
20
LBL
RTO
RETENTIVE TIMER ON
Timer T4:3
Time base (sec) 1.0
Preset 50
Accumulated 0
EN
DN
T4:3/DN
T4:3
RES
子路径文件3
SBR
SUBROUTINE
PL3
12
PL5
TON
TIMER ON DELAY
Timer T4:6
Time base 1.0
Preset 5
Accumulated 0
RET
RETURN
输出
L2

图 9-29

第 10 章 数据处理指令

学习目标：
- 定义数据处理并通过编写 PLC 程序学会运用。
- 描述字级指令从一个存储区到另一个存储区复制数据的工作过程。
- 在 PLC 应用中解释数据传输和数据比较指令。
- 比较离散 I/O、多位数字 I/O 和模拟量 I/O。
- 描述闭环控制系统的工作原理。

数据处理包括下面两个方面的内容：（1）数据传输；（2）对数据进行数学运算、转换、比较和逻辑操作。本章介绍的数据处理指令，既有关于字数据的，也有关于包含多个字的文件数据。数据处理在 PLC 内部的执行方式与在计算机上的执行方式类似。最后，本章讨论了需要这些操作的一个过程实例。

10.1 数据处理

数据处理指令使 PLC 具备了计算机系统的一些功能。目前大多数的 PLC 具有处理存储在内存中数据的能力。这使得 PLC 的功能远远超过了传统的与继电器相当的指令功能。

数据处理涉及数据传输和对数据进行数学运算、转换、比较和逻辑操作。实现这个功能的指令基本可以分为两类：在字数据上面进行操作和在包含多个字的文件或块文件上面的操作。

每个数据操作指令都需要两个或多个数据存储字进行操作。根据不同的生产厂家，单个的数据存储字可以是寄存器也可以是字。当一组连续的相关数据存储器被引用时，一般采用表格（table）或文件（file）。图 10-1 说明了字和文件之间的区别。保存在文件和字中的数据以二进制的形式存放，以一连串的 1 和 0 来表示。在 Allen-BradleySLC-5 PLC 中一组连续的元素或字被称为文件，而在 ControlLogix 控制器中称为数组。

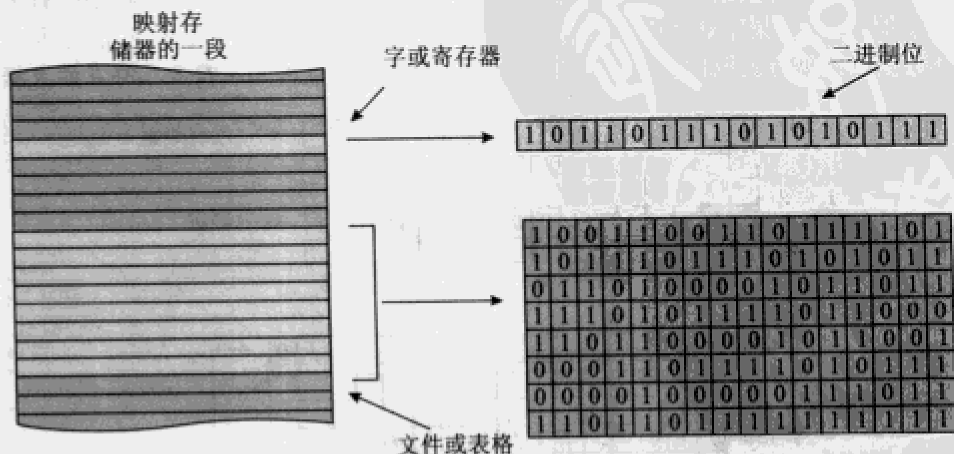


图 10-1 数据文件、字和位

数据处理指令允许在 PLC 中的一个存储区和另一个存储区之间，对单字或多字组进行数据移动、处理和存储。在需要产生并处理大量数据的实际应用中，使用这些数据处理

指令，能够极大地减小PLC程序的容量，并降低复杂性。数据处理可以分为两大类：**数据传送和数据比较**。

267
↓
268

对整个字的处理功能是PLC的一个重要特征。它使PLC能够处理包含多位信息的输入输出结构，如模拟量输入和输出模块。算术运算也需要PLC内部的数据，以字或寄存器的形式处理。为了方便解释各种不同的数据处理指令，以Allen-Bradley PLC-2、PLC-5和SLC-500PLC的指令为例进行说明。尽管各个厂家的PLC指令及其形式各不相同，但是数据处理的概念是一样的。图10-2所示为基于Allen-Bradley SLC-500PLC及其编程软件RSLogix的典型数据处理指令。



命令	名　称	说　明
MOV	传送指令（Move）	将数据从源地址移动到目的地址
MVM	屏蔽传送指令（Masked Move）	将数据从源位置传送到目的地址，并且允许部分目的数据被一个独立的字屏蔽
AND	与（And）	逐位进行与操作
OR	或（Or）	逐位进行或操作
XOR	异或（Exclusive Or）	逐位进行异或操作
NOT	非（Not）	非操作
CLR	清零（Clear）	把字的各个位都置0

图10-2　基于Allen-Bradley SLC-500PLC及其编程软件RSLogix的数据处理指令

10.2　数据传送操作

数据传送指令就是把数据从一个字或寄存器传送到另一个字或寄存器。图10-3所示为将二进制数据从存储器的一个位置传送到另一个位置。图10-3a显示数字信息存储在字130中，并且字040中没有数据。图10-3b显示出当数据传送操作完成后，字040中就有了与字130中相同的数据。如果字040中已经存在了其他的数据（不全为0），这些信息也会被替代。当新数据以这种方式取代了已经存在的数据，这个过程被称为***写覆盖已存在的数据***。

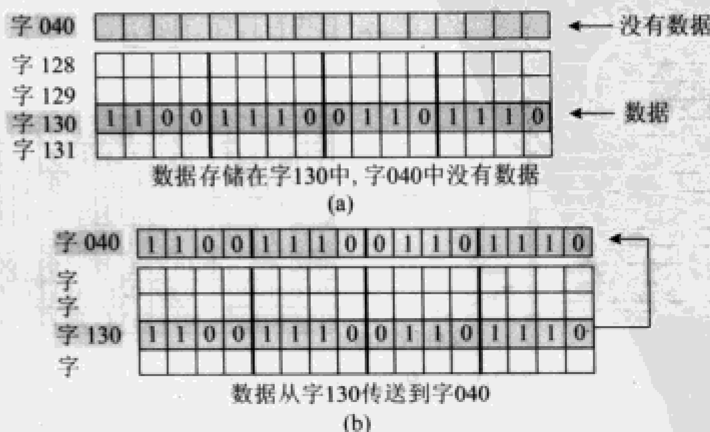


图10-3　数据传送

数据传送指令几乎可以寻址存储器中的任何位置。预先存储的数据可以自动的释放并放置于任何一个新的位置。这个新的位置可以是定时器或计数器的预置寄存器，甚至可以是驱动七段显示管的输出寄存器。 269

通常有两种基本类型的指令用于获取和传送数据。有些厂家采用一个单独的指令实现整个操作，而有些厂家采用两个独立的指令。Allen-Bradley PLC-2 控制器采用线圈形式的数据传送指令：GET 和 PUT。GET 指令告诉处理器要获取存储在其他字中的数据。GET 指令编程时放在梯级中的状态位置。它一直为一个逻辑真指令，能够从数据表中的特定位置获得一个 16 位的字。

PUT 指令告诉处理器把由 GET 指令获得的数据放在什么位置。编程时，PUT 指令放在梯级的输出位置。PUT 指令立即接收来自前面 GET 指令的数据。它用来存储其他操作的结果，存储位置由 PUT 线圈确定。

PUT 指令与 GET 指令一起构成数据传送梯级。图 10-4 所示为一个数据传送梯级的例子。当输入 A 为真时，GET/PUT 指令告诉处理器取出储存在字 020 中的数值 005，并把它放入字 130 中。任何情况下，PUT 指令的前面都要有 GET 指令。 270



图 10-4　GET/PUT 指令的数据传送梯级

Allen-Bradley PLC-5 和 SLC-500 控制器，采用模块形式的传送指令（MOV）来完成数据传送。MOV 指令把数据从一个字复制到另一个字，从源地址到目的地址。图10-5所示为 MOV 指令的一个例子。在这个例子中，当梯级条件为真时，存储在源地址 N7:30 中的数据，被复制到目的地址 N7:20 中。当梯级条件变为假时，目的地址中的数据不变，除非在程序中的其他地方被改变。这个指令在编程时前面可以有输入条件，也可以没有输入条件。



图 10-5　PLC-5 和 SCL-500 模块形式的传送指令

屏蔽传送指令（MVM）与传送指令（MOV）指令有些区别，在屏蔽传送过程中包含了一个屏蔽字。要传送的数据，必须通过屏蔽字才能到达目的地址。MVM 指令用于传送 16 位字中想要的部分，而屏蔽其余的部分。

图 10-6 所示为屏蔽传送指令（MVM）的一个例子。指令传送存储在源地址 B3:0 中

的数据，通过屏蔽字到达目的地址 B3:4。屏蔽字可以以地址或十六进制的形式输入，它的值以十六进制的形式显示。当屏蔽字的位为 1 时，相应的源地址数据位将传送到目的地址。相反，当屏蔽字的位为 0 时，目的地址相应位的数据保持不变。屏蔽字的大小与源地址和目的地址相同。

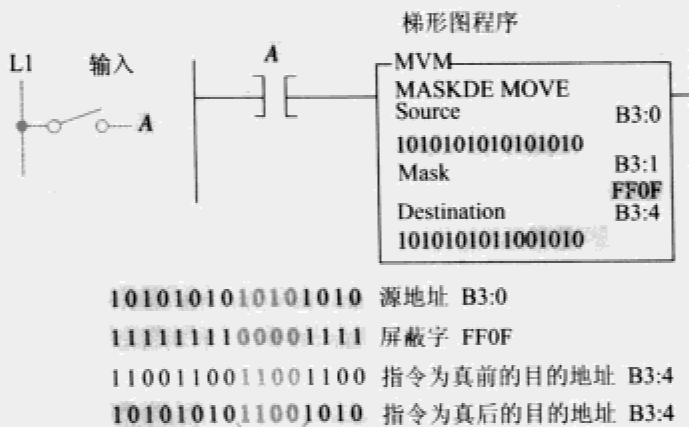
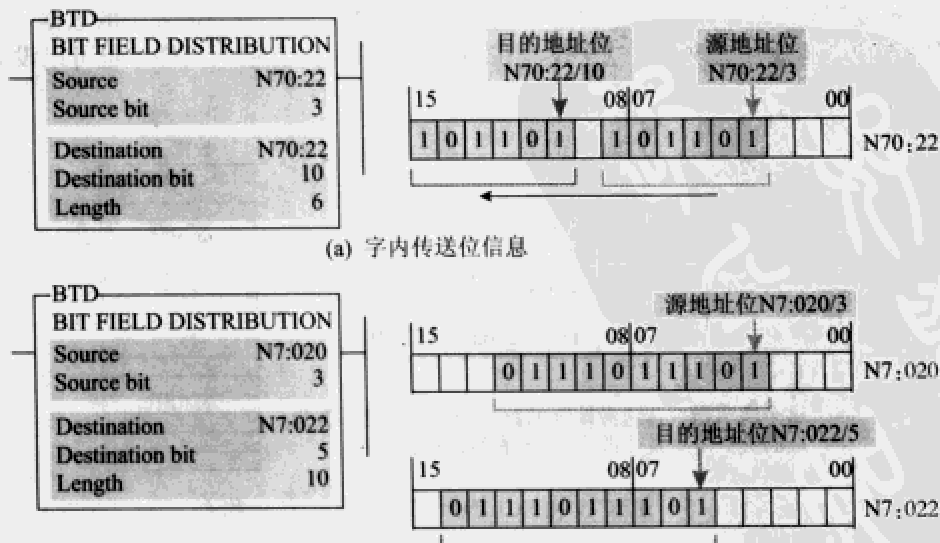


图 10-6　屏蔽传送指令（MVM）

位域分配指令（BTD）用于在一个字内或字之间传送数据位，如图 10-7 所示。在每个扫描周期，当包含了 BTD 的梯级条件为真时，处理器从源字的位域移动数据到目的字。如果是在一个字内移动数据，源地址字和目的地址字相同。源数据没有发生改变，目的位置的特定位被覆盖。



(a) 字内传送位信息

(b) 字间传送位信息。当超过了目的字地址的范围时数据会丢失，不会传送到下一个更高位的字

图 10-7　位域分配指令

图 10-8 所示为利用 MOV 数据传送指令改变通延时定时器预置时间的应用实例。当选择开关在 10s 位置时，梯级 2 接通，梯级 3 断开。存储在源地址 N7:1 中的数据 10 被复制到目的地址 T4:1 中，因此定时器 T4:1 的预置值从 0 变为 10。当按钮 PB1 闭合时，经过 10s 的延时，指示灯亮。要想把定时器的预置时间改为 5s，选择开关放在 5s 的位置，这时梯级 3 接通而梯级 2 断开。定时器 T4:1 的预置值从 10 变为 5。按钮 PB1 闭合时，经过 5s 的延时，指示灯亮。

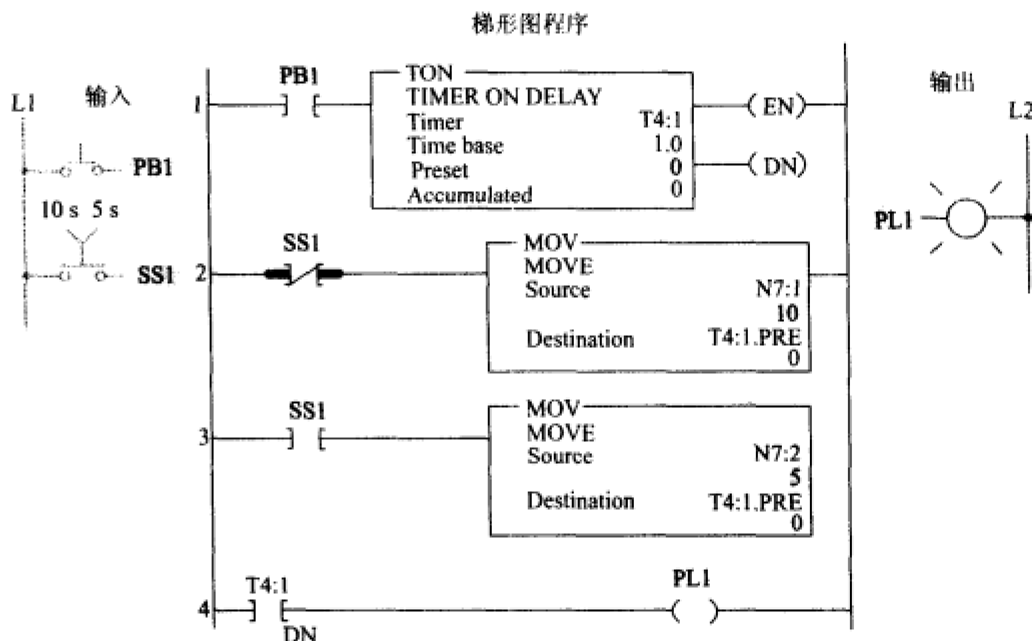


图 10-8 使用传送指令（MOV）改变定时器的预置值

图 10-9 所示为使用 GET/PUT 指令和 MOV 数据传送指令改变加计数器预置值的例子。在本例中，用一个限位开关来控制计数器，该计数器对从传送带到存储机架上产品进行计数。生产线上可以有三种不同类型的产品：在存储架可以存储 300 件的产品 A、175 件的产品 B 和 50 件的产品 C。根据生产线上传输的不同类型的产品（A、B 或 C），可以用三个开关选择想要的计数器预置值。用一个复位按钮对累加计数值清 0。当产品存满存储机架时，指示灯会指示。如果不止一个选择预置计数值的开关闭合，只选定最后一个值。

文件是在数据表格中一组相关的连续字，有确定的开始和结束，用于存储信息。例如，在批处理程序中，操作员可以选择包含在不同文件中的许多独立的配置程序。

在有些情况下，需要把整个文件，从 PLC 存储器的一个位置移动到另一个位置。这种文件转移被称为文件到文件的转移。当一个文件中的数据代表了一系列的状态，这些状态在 PLC 的程序中又相互作用，而且要保证每次操作后这些状态不发生改变的时候，就要使用文件到文件的转移。

因为包含在这个文件中的数据，一定会因为程序的作用而发生改变，所以用另一个文件处理这些改变的数据，并且另一个文件中的数据允许被程序修改。而包含在第一个文件中的数据保持不变，因而就可以多次使用。其他类型的文件数据处理指令，还包括字到文件和文件到字的移动指令（如图 10-10 所示）。

271～272
273

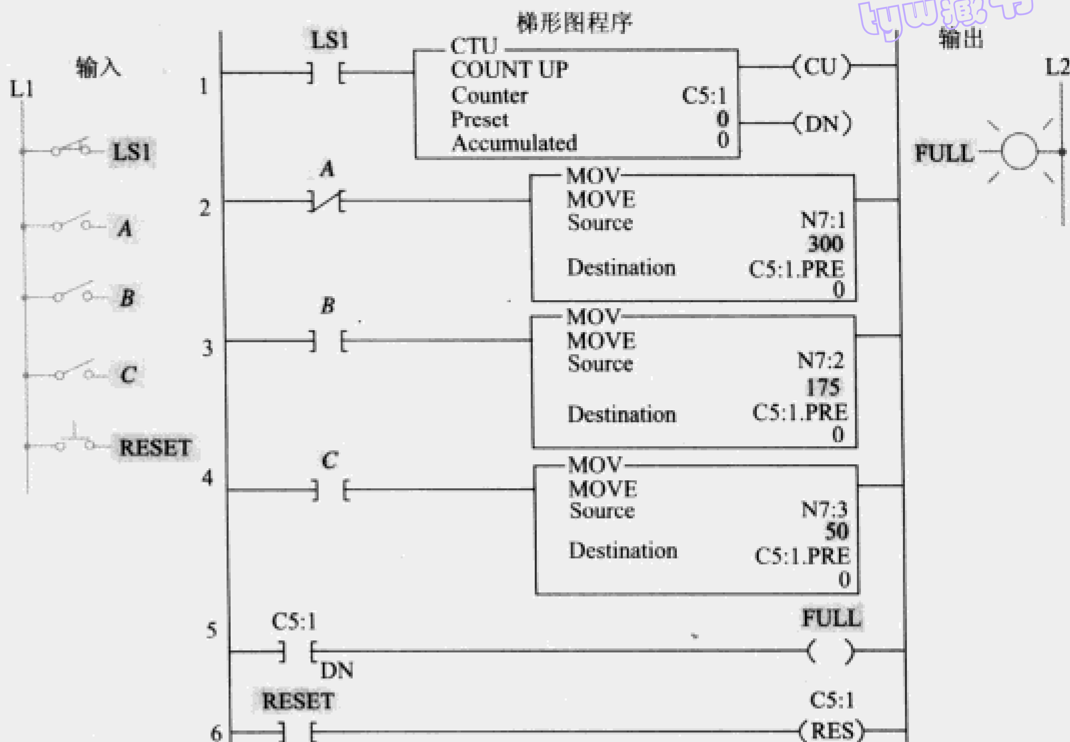


图 10-9　利用数据传送指令（MOV）改变计数器的预置值

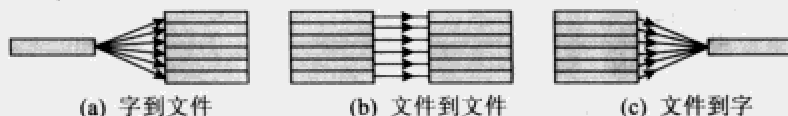


图 10-10　使用文件指令移动数据

文件可以使大量的数据被快速地扫描，而且对需要对数据进行转移、比较或转换的编程也非常有用。大多数的 PLC 生产厂家都把文件指令以模块的形式显示在编程终端显示器上。Allen-Bradley PLC-5 和 SLC-500 控制器，用前缀#来表示文件的开始，如图 10-11 所示。#在一个字或元素地址中被省略。

274

文件运算与逻辑指令（FAL）用于把数据从一个文件复制到另一个文件，并且对文件进行数学和逻辑操作。FAL 指令如图 10-12 所示，这条指令的基本操作跟所有的函数类似，需要在指令中输入下列的参数和地址：

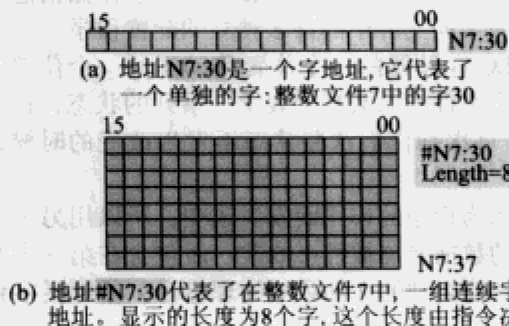


图 10-11　Allen-Bradley PLC-5 和 SLC-500 的字与文件地址

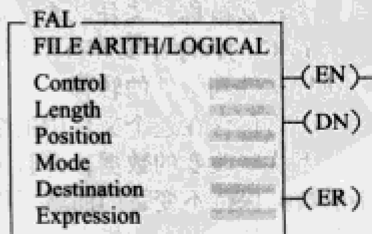


图 10-12　FAL 指令

- **Control**（控制）是第一个输入参数，它是在处理器存储区控制结构体的地址。处理器利用这个信息来运行这个指令。默认的控制文件是数据文件6。其他的控制文件可以是从文件9到文件999。FAL的控制元素必须是唯一的，不能再用它去控制其他的指令。控制元素由三个字组成。在FAL指令中的控制字使用四个控制位：位15（使能位），位13（完成位），位11（错误位）和位10（卸载位）。

位：	15 13 11 10
控制字：R6:0	EN DN ER UL
字长度：R6:0. LEN	存储文件长度 1－1 000
位置字：R6:0. POS	在文件中的位置

- **Length**（长度）（控制元素的第二个字，有时是第三个字）。它是指令的第二个参数，代表了文件的长度。除了浮点数文件外（浮点数包含两个字），它以字的形式输入。最大的可能长度为1 000，可以输入1到1 000之间的任何十进制数。
- **Position**（位置）（控制元素的第四个字）。它是第三个输入参数，代表了当前处理器正在访问的数据块的位置。它指出了正在被操作的字，位置从0开始，一直索引到比文件长度小1。通常在文件的起点输入0来开始操作，也可以输入一个让想要的FAL指令开始执行的位置。当指令被复位时，位置也被置为0。可以在程序中对位置进行处理。
- **Mode**（模式）。第四个输入参数，代表了每个扫描周期中被执行的文件号。有三种选择：

  275

  - 输入A选择*全模式*。在全模式下，指令在一个扫描周期中传送数据的整个文件。当指令接通时，使能位（EN）为真，并延续梯级条件。当所有的数据被传送完，完成位（DN）变为真。在同一个扫描周期下，当指令变为真时，也会发生这种改变。由于在数据传输过程中发生错误指令而不能完成扫描时，如存储的数据相对于数据表类型太大或太小时，指令将会在那个位置停止，并把错误位（ER）置位。扫描继续进行，但指令不会继续执行，直到ER位被复位。当指令执行完成，在指令没有断开之前使能位和完成位一直被置位，指令变为假时，使能位和完成位都被复位为0。
  - 输入1－1 000的十进制数选择*数字模式*。在数字模式下，文件操作被分为多个程序扫描完成。输入的数字值，决定了每个周期要传送的数据量。数字模式下，可以减少完成程序扫描的时间。不是在一个扫描周期等待把所有的文件长度都传送完，而是把文件数据传送分为多个扫描周期完成，因此降低了每个扫描周期指令执行的时间。
  - 输入I选择*增量模式*。在增量模式下，指令梯级的每个上升沿到来时，单元数据被进行操作。上升沿第一次到来时，位置（position）为0，文件数据的第一个单元被操作。位置一直保持为0，UL位被置位，EN位也跟随指令的状态。上升沿第二次到来时，位置（position）变为1，文件中的第二个字的数据被操作。无论指令要操作当前位置的数据，还是要索引位置然后再传输数据，UL位都可以控制。如果UL位处于复位状态，当梯级上升沿到来，指令执行当前位置的操作，并置位UL位；如果UL已经置位，上升沿到来时，位置变为1，并且在新的位置下对数据进行操作。
- **Destination**（目的地址）。指令的第五个输入参数，它是处理器存储操作结果的地址。指令把数据转换为由目的地址确定的数据类型。它可以是文件地址，也可以是单元地址。

- **Expression**（表达式）。最后一个输入参数，它包含了能确定数据源和执行操作的地址、程序常数和运算符。表达式决定了 FAL 指令的功能。它可以包含文件地址、单元地址或者一个常数，也可以只是一个函数，因为 FAL 指令也可以只担当一个函数的功能。

图 10-13 所示为使用 FAL 指令实现文件到文件复制功能的实例。当输入 A 闭合，存储在开始地址为#N7:20 的表达式文件中的数据，被复制到开始地址为#N7:50 的目的文件。两个文件的长度由控制单元字 R6:1. LEN 的输入值决定。在这个例子中的指令使用了全模式，也就是说当 FAL 指令遇到上升沿时，所有的数据在第一个扫描周期被传送。如果在数据传输过程中没有错误发生，DN 位会被置位；当有错误发生时，ER 位就会被置位，指令在该位置停止运行，程序继续扫描下一条指令。

276

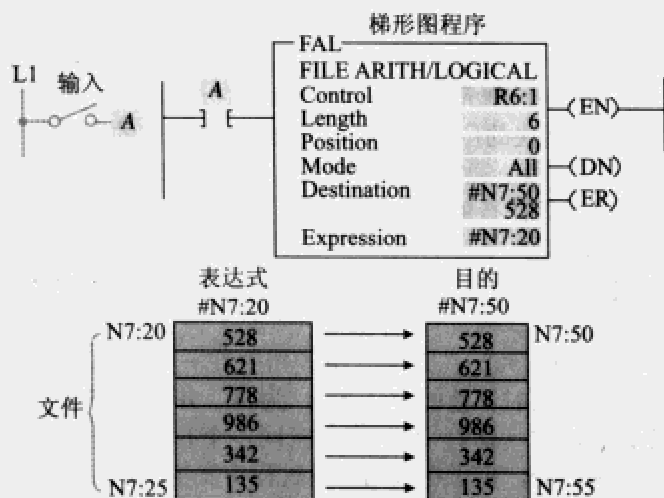


图 10-13　使用 FAL 指令实现文件到文件的复制功能

图 10-14 所示为用 FAL 指令实现文件到字复制功能的例子。当输入 A 遇到上升沿时，处理器读取整数文件 N29 中的一个单元，从单元 0 开始，并且把数据映像写入到文件 N29 的第 5 个单元。在目的地址的任何数据都被覆盖。

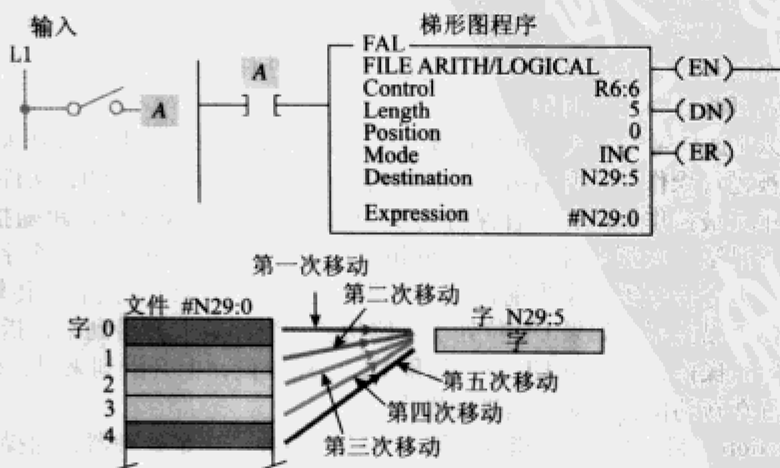


图 10-14　用 FAL 指令实现文件到字的复制功能

图 10-15 所示为使用 FAL 指令实现字到文件复制功能的实例。除了把数据从字地址复制到文件外，其他的都与文件到字的复制功能类似。注意表达式为一个字地址（N7:100），目的是一个文件地址（#N7:101）。如果从位置 0 开始，在输入 A 的第一个上升沿数据从 N7:100 复制到 N7:101。在第二个上升沿，复制 N7:100 的数据到 N7102。每一个上升沿的到来，数据都被复制到文件的下一个位置，直到文件的最后一个单元 N7:106。 277

对于 FAL 指令中的定时器、计数器和控制数据文件来说，在数据表中必须包含文件地址的规则是个特例。在这些数据文件中，如果要指定文件地址，FAL 指令要在对应文件类型的文件中，取三个字作为预置值、累加值、长度或位置数据文件。如图 10-16 所示，为存储了定时器预设值的数据配置被传送到定时器的预置值文件。

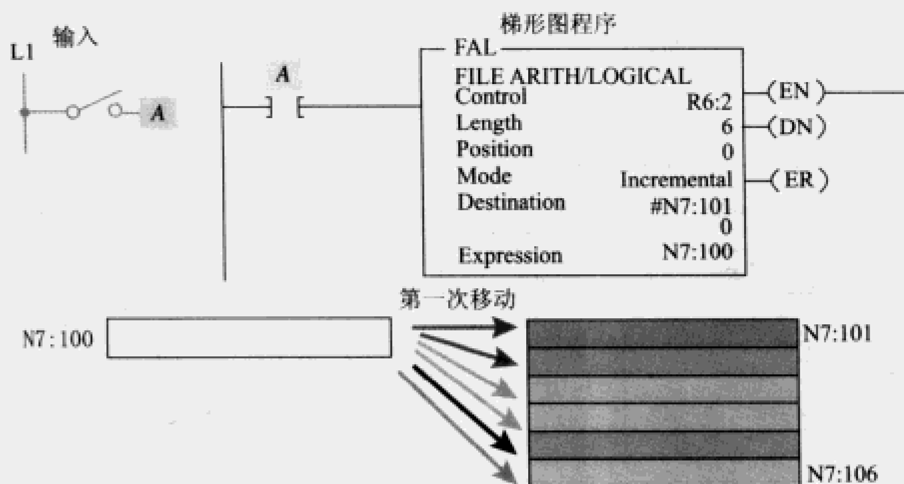


图 10-15 用 FAL 指令实现字到文件复制功能

文件复制（COP）指令与文件填充（FLL）指令是高速处理指令，比同样操作的 FAL 指令执行要快。与 FAL 指令不同的是，它没有作为监控和处理的控制单元，没有数据转换发生，因此源地址和目的地址具有相同的文件类型。使用 COP 指令的一个例子如图 10-17 所示，源地址和目的地址都是文件地址，当输入 A 闭合时，文件 N40 中的数据被复制到文件 N20。在指令接通的每个扫描周期，指令都复制文件的长度。 278

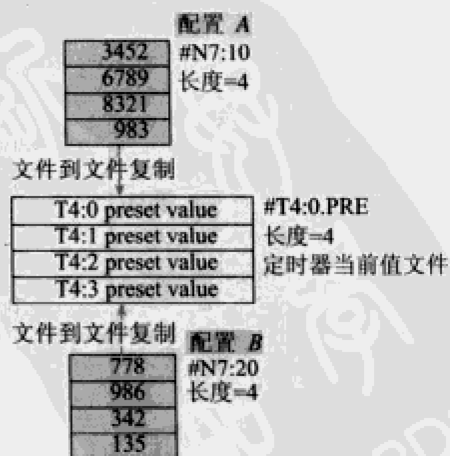


图 10-16 定时器预置值复制配置数据并存储

图 10-18 所示为使用填充文件（FLL）指令的例子。它与工作在全模式下执行字到文件复制功能的 FAL 指令非常相似。当输入 A 闭合，存储在 N15:15 中的数据被复制到从 N20:1 到 N20:6 的字中。因为指令把数据传送到文件的最后单元，因此每个字中的数据都相同。 279

FLL 指令常常用于把一个文件的所有数据清 0，如图 10-19 所示的程序。按下按钮 PB1，将文件#N10:0 中的数据复制到文件#N12:0。按下 PB2，文件#N12:0 中的数据被清 0。注意源地址中的数据为 0。

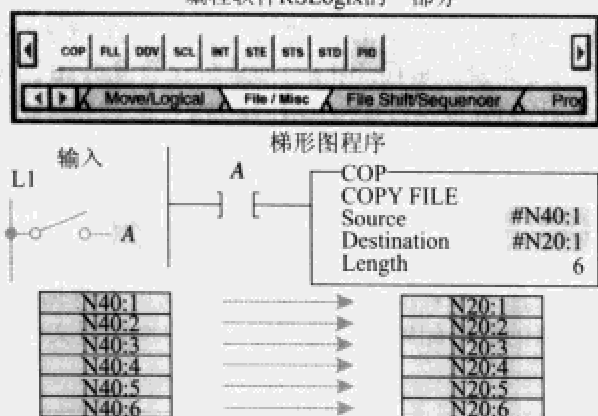


图 10-17　文件复制（COP）指令

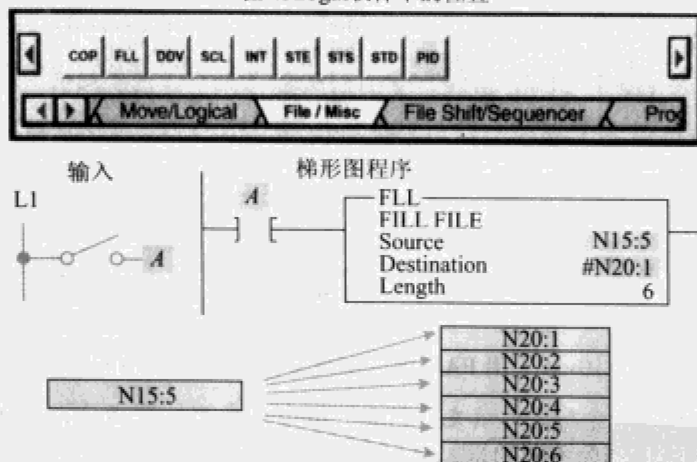


图 10-18　文件填充（FLL）指令

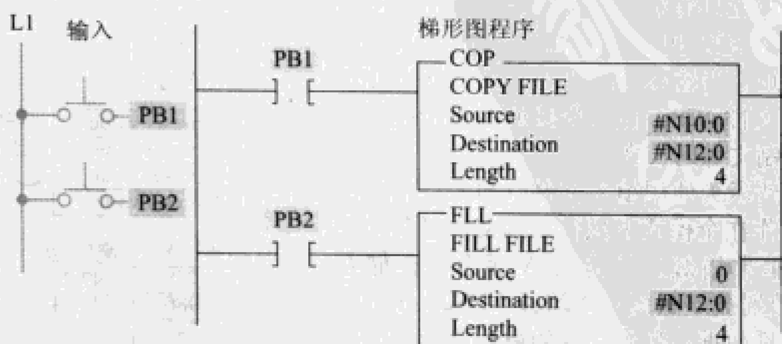


图 10-19　使用 FLL 指令对文件中所有数据清 0

10.3　数据比较指令

数据传送指令都是输出指令，而数据比较指令属于输入指令。字比较指令在工业中用

得非常多，它用于在下一步处理之前，需要对数据进行比较的应用场合。数据比较指令的应用包括以下几个方面：

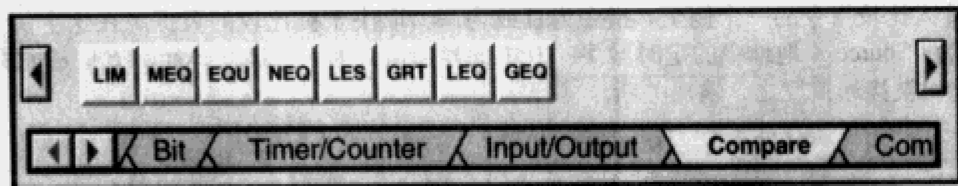
- 当计数器达到特定值时，开始一个操作或过程。
- 输入设备的数据（指轮开关）在发送到定时器或计数器的预置值之前，保证数据在正常范围内。
- 检验数据在允许的范围内。

数据比较指令比较存储在两个或多个字（寄存器）中的数据，并根据程序指令做出决定。根据不同的 PLC，存储在存储器两个字中的数据可以进行下列状态的比较。

名称	符号
小于（LESS THAN）	（<）
等于（EQUAL TO）	（=）
大于（GREATER THAN）	（>）
小于等于（LESS THAN OR EQUAL TO）	（≤）
大于等于（GREATER THAN OR EQUAL TO）	（≥）

数据比较的概念已经用于定时器和计数器指令上面，在前面第 7 章和第 8 章中已做了解释。在这两个指令中，当定时器或计数器的累加值达到预置值时，接通或关断输出。实际上就是在每一个处理器扫描周期，将一个存储字中的累加数据与另一个存储字中的预置值相比较。当处理器判断出累加值与预置值相等时，控制输出接通或关断。 280

比较指令用于比较两个数据，从而决定梯级是否为真。图 10-20 所示为基于 Allen-Bradley SLC-500 PLC 和它的编程软件 RSLogix 的典型数据比较指令。



命令	名称	说明
LIM	极限比较指令	测试一个数据是否在另外两个数据之间
MEQ	相等屏蔽比较	测试两个数据的部分数据位是否相等。通过屏蔽字比较源地址的 16 位数据和参考地址的 16 位数据
EQU	等于	测试两个数据是否相等
NEQ	不等于	测试一个数据是否不等于另一个数据
LES	小于	测试一个数据是否小于另一个数据
GRT	大于	测试一个数据是否大于另一个数据
LEQ	小于等于	测试一个数据是否小于或等于另一个数据
GEQ	大于等于	测试一个数据是否大于或等于另一个数据

图 10-20　基于 Allen-Bradley SLC-500PLC 和它的编程软件 RSLogix 的典型比较指令

等于（EQU）指令是一种用于比较 Source *A* 和 Source *B* 的输入指令：当 Source *A* 等于 Source *B* 时，指令的逻辑变为真；反之，如果不等于时变为逻辑假。图 10-21 所示为 EQU 输入比较指令的例子。当存储在 Source *A* 中的定时器 T4:0 中的预置值，等于存储在 Source *B* 中 N7:40 中的数据时，指令为真，输出被激励。Source *A* 可以是字地址，也可以是浮点数地址。Source *B* 可以是一个字地址、浮点数地址或是一个常数值。因为精度的需要，不推荐浮点数采用等于指令，优先采用数据极限比较指令。 281

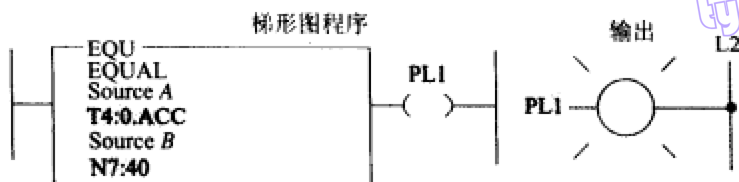


图 10-21 EQU 指令逻辑梯级

不等于（NEQ）指令是一种用于比较 Source *A* 和 Source *B* 的输入指令：当 Source *A* 不等于 Source *B* 时，指令的逻辑变为真；反之，如果等于时变为逻辑假。图 10-22 所示为使用 NEQ 输入比较指令的一个例子。当存储在 Source *A* 的地址 N7:5 中的数据不等于 25 时，输出为真；反之，如果等于时，输出为假。在所有的输入比较指令中，Source *A* 和 Source *B* 中的内容可以是数据，也可以是包含了数据的地址。

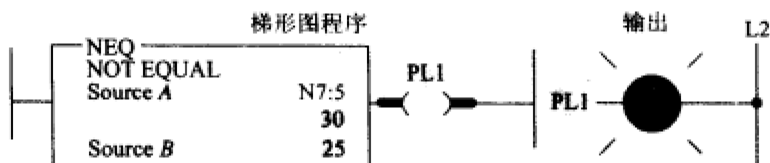


图 10-22 NEQ 指令逻辑梯级

大于（GRT）指令是一种用于比较 Source *A* 和 Source *B* 的输入指令：当 Source *A* 大于 Source *B* 时，指令的逻辑变为真；反之，如果不大于时变为逻辑假。图 10-23 所示为 GRT 输入比较指令的一个例子。指令为真或为假，取决于被比较的数据。在这个例子中，当存储在 Source *A* 地址中的定时器 T4:10 中的累加值，大于存储在 Source *B* 中的常数 200 时，输出被接通；反之，输出关断。

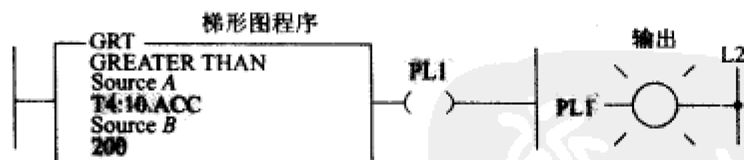


图 10-23 GRT 指令逻辑梯级

小于（LES）指令是一种用于比较 Source *A* 和 Source *B* 的输入指令：当 Source *A* 小于 Source *B* 时，指令的逻辑变为真；反之，如果不小于时变为逻辑假。图 10-24 所示为 LES 输入比较指令的一个例子。在这个例子中，当存储在 Source *A* 的地址中的计数器 C5:10 的累加值，小于存储在 Source *B* 中的常数 350 时，输出被接通，反之输出被关断。

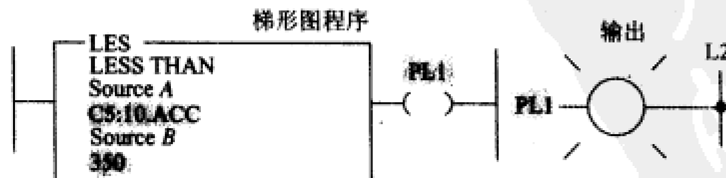


图 10-24 LES 指令逻辑梯级

大于等于（GEQ）指令是一种用于比较 Source *A* 和 Source *B* 的输入指令：当 Source *A* 大于或等于 Source *B* 时，指令的逻辑变为真；反之，如果小于时变为逻辑假。图 10-25 所示为 GEQ 输入比较指令的一个例子。当存储在 Source *A* 的地址 N7:55 中的数据，大于或

等于存储在 Source *B* 中的地址 N7:12 中的数据时，输出被接通，反之输出被关断。

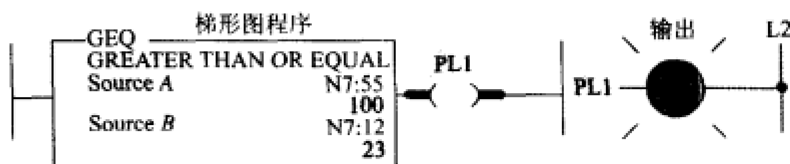


图 10-25 GEQ 指令逻辑梯级

小于等于（LEQ）指令是一种用于比较 Source *A* 和 Source *B* 的输入指令：当 Source *A* 小于或等于 Source *B* 时，指令的逻辑变为真；反之，如果大于时变为逻辑假。图 10-26 所示为 LEQ 输入比较指令的一个例子。在这个例子中，如果计数器 C5:1 的累加值小于或等于 457 时，输出被接通，反之输出被关断。

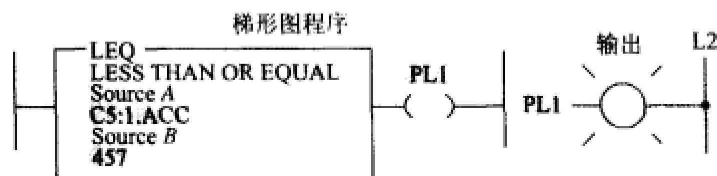


图 10-26 LEQ 指令逻辑梯级

282
283

如图 10-27 所示的极限比较（LIM）指令，用于对测试值与高限位值和低限位值比较。这个指令被称为是全集指令，因为它可以工作在下列两种方式：

- 高限位值大于低限位值时，如果测试值在两个限位值之间或者与限位值相等，指令为真。
- 低限位值大于高限位值时，如果测试值大于等于低限位值或者小于等于高限位值时，输出为真。

在图 10-27 a 中，高限位值为 50，低限位值为 25。当测试值在 25 和 50 之间时，指令为真。因为图中的测试值为 48，所以指令输出为真。如图 10-27b 所示，高限位值为 50，低限位值为 100。当测试值小于等于 50 或大于等于 100 时，输出为真。因为图中的测试值为 125，所以输出为真。使用极限比较指令的应用包括：

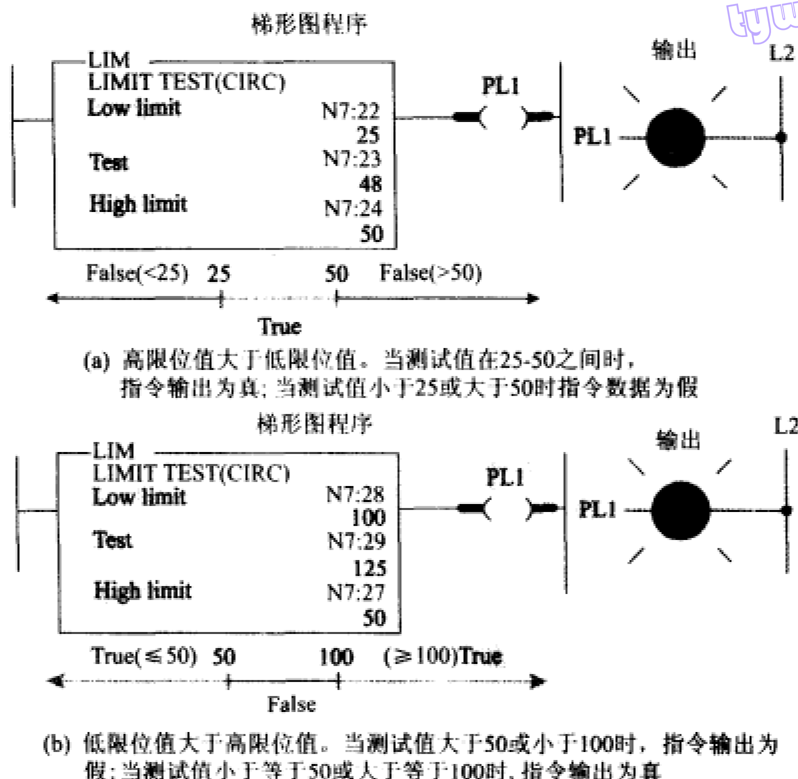
- 当温度在设定范围内，启动搅拌机。
- 当温度在设定范围之外，允许控制过程发生。

284

图 10-28 所示为相等屏蔽比较（MEQ）指令输入比较指令的例子。这个指令用于比较存储在 Source 地址中的数据和存储在 Compare 地址中的数据，并且允许屏蔽部分数据。如果在 Source 地址中的数据与在 Compare 地址中的数据相等（没有被屏蔽的位），指令为真；一旦检测到一个不相等的位，指令就变为假。当屏蔽位为 1 时数据通过屏蔽字，为 0 时，数据被阻止。屏蔽字的大小必须与 Source 和 Compare 的地址相同（16 位）。屏蔽字的位被设置为 1 时，相应位的数据才能进行比较；当屏蔽字中的位设为 0 时，不会对相应的 Compare 地址中的数据进行比较。如果要在梯形图程序中改变屏蔽字中的数据，可以把屏蔽字存储在一个数据地址中。否则，可以直接输入 16 进制数据作为屏蔽字。

使用相等屏蔽比较指令的应用包括：

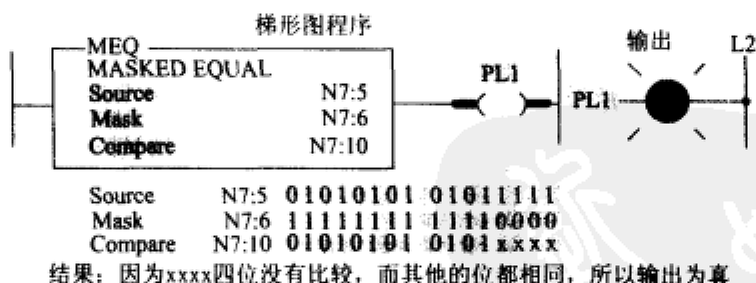
- 比较 16 个限位开关是否达到期望位置。Source 中存储的是限位开关的地址，Compare 中存储的是期望的状态。屏蔽字可以阻止不用比较的限位开关状态。
- 一个 2 位指轮开关连接到输入模块，如果要这 8 位状态与期望值比较，可以把接到模块的其他输入屏蔽掉，只对 8 位进行比较。



(a) 高限位值大于低限位值。当测试值在25-50之间时，指令输出为真；当测试值小于25或大于50时指令数据为假

(b) 低限位值大于高限位值。当测试值大于50或小于100时，指令输出为假；当测试值小于等于50或大于等于100时，指令输出为真

图 10-27 极限比较指令（LIM）的逻辑梯级



结果：因为xxxx四位没有比较，而其他的位都相同，所以输出为真

图 10-28 相等屏蔽比较指令（MEQ）逻辑梯级

10.4 数据处理程序

数据处理指令增加了控制电路编程的尺度和灵活性。例如，图 10-29 所示的原始继电器延时电路。在这个电路中，使用了三个气动延时继电器控制四个电磁阀。当按下启动按钮时，电磁阀 *A* 立即接通，5s 后电磁阀 *B* 接通，10s 后电磁阀 *C* 接通，15s 后电磁阀 *D* 接通。

285

这个电路可以用传统的 PLC 程序和三个内部定时器实现。然而这个电路也可以仅使用一个内部定时器和数据比较指令实现，这将导致存储器字的保存。图 10-30 所示为用一个内部定时器实现的程序。假设停止按钮已经闭合，当按下开始按钮时，SOL *A* 线圈被接通，SOL *A* 的触点闭合并把线圈自锁，同时 SOL *A* 的触点接通延时定时器 T4:1。定时器

的预置值为 15s。当 15s 延时到时，定时器的完成位 T4:1/DN 置位，使输出 SOL *D* 接通。当定时器延时 5s 后，即累计的时间等于并超过了 5s，输出 SOL *B* 接通。相同的，10s 后，即累计的时间等于并超过了 10s，输出 SOL *C* 接通。

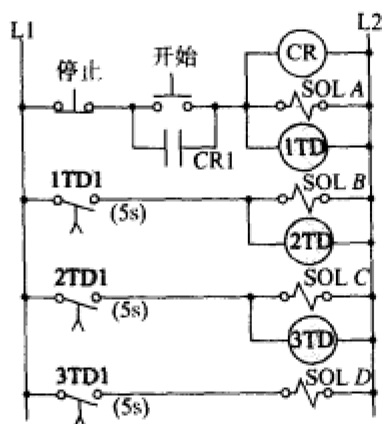


图 10-29 原始继电器延时电路

图 10-31 所示为利用等于指令的通延时定时器程序。当开关（S1）闭合时，定时器 T4:1 开始定时。这两个等于指令的 Source *A* 地址中存储的都是定时器运行时的累加值。梯级 2 中的等于指令 Source *B* 中存储的值为 5。当定时器的累加值达到 5 时，接通指令并持续 1s，锁存输出被激励，指示灯 PL1 接通。当定时器的累加值达到 15s 时，接通梯级 3 中的等于指令并持续 1s，解锁输出被激励，指示灯 PL1 关断。因此当开关闭合时，5s 后指示灯接通，保持 10s，然后关断。

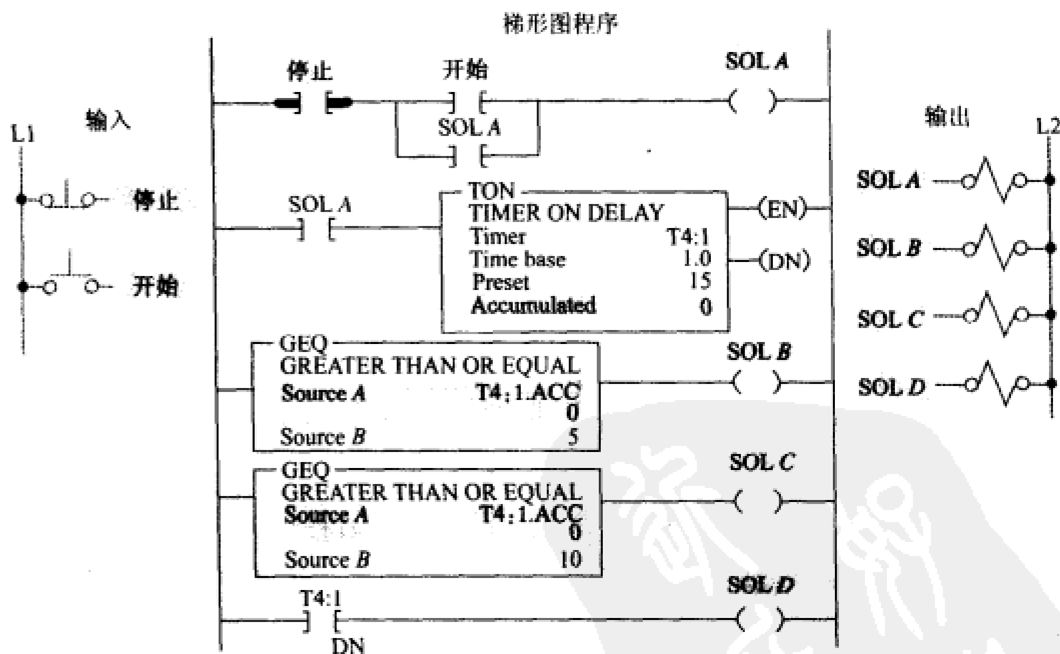


图 10-30 使用 GEQ 指令实现多定时器功能

286

图 10-32 所示为采用了小于指令的加计数器程序。每次按钮 PB 的上升沿到来时，加计数器 C5:1 加 1。小于指令的 Source *A* 中存放的是计数器的累加值，Source *B* 中存储常数 20。只要 Source *A* 中的数据小于 Source *B* 中的数据，小于指令为真。因此，当计数器累加值在0－19之间时，输出和指示灯 PL1 接通。当计数器的累加值到达 20 时，小于指令变为假，关断输出和指示灯。当计数器的累加值达到它的预置值，计数器被复位，累加值清 0。

数据比较指令一般可以直接使用。但是当在 PLC 的程序中使用这些指令控制容器的材料填充时，常会出现编程错误。接收容器在被填充时，通过 PLC 程序对重量进行连续监控。当重量达到预设值时，填充被切断。填充过程中，PLC 的处理器比较容器的当前值和预设值。如果程序中仅仅使用了等于指令，就会出现问题。刚填充时，比较指令为假。

287

当容器重量达到期望值时，等于指令变为真，流量被停止。但是，系统会渗漏一些多余的材料到容器中，容器的总重量就会超过预设值，指令又变为假，引起容器的过充。解决这个问题的最简单的方法，是在程序中用大于等于指令取代等于指令。这样任何多余的材料都不会引起又一次的填充操作。然而这种情况下，有必要增加额外的编程，来监视严重的满充状态。

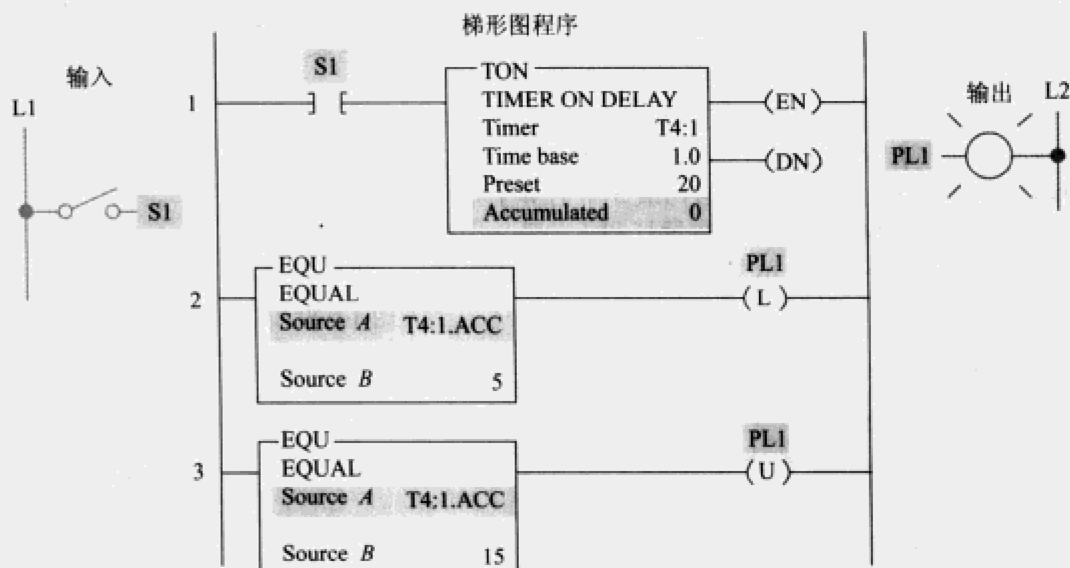


图 10-31 使用 EQU 的定时程序

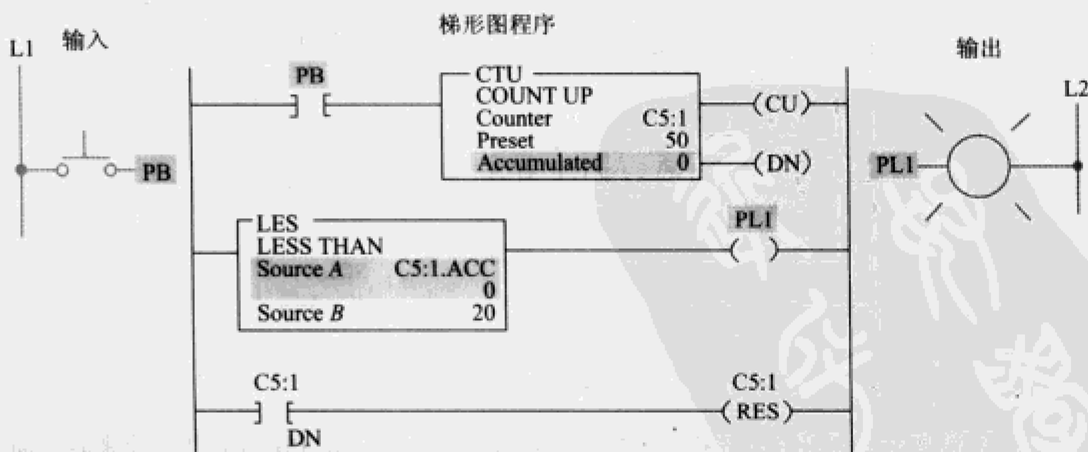


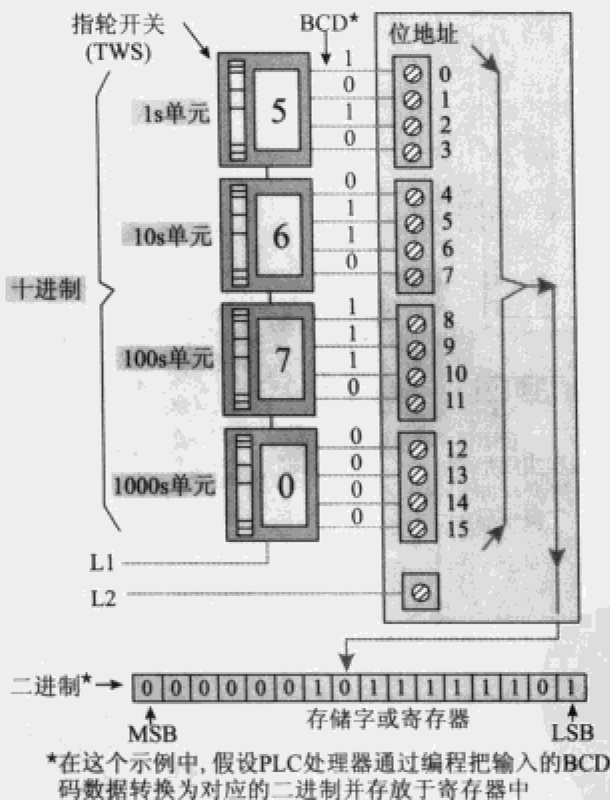
图 10-32 使用 LES 的计数器程序

10.5 数字信号 I/O 接口

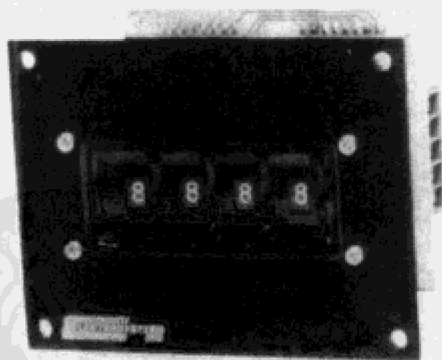
PLC 不断扩大的数据处理能力，导致了一种新型的 I/O 接口设备——数字信号 I/O 接口。一般的，数字信号 I/O 接口可以分为两类：一种是为多位数字设备提供接口，另一种是为模拟量设备提供接口。

多位数字设备类似于离散 I/O 接口，因为它们处理的信号都是离散的（通/断）。它们之间的区别是：离散 I/O 一次只读取一位输入或控制一位输出；而多位接口允许多个位作为一个单元输入或输出。它们用于配合那些需要 BCD 输入或输出的设备。

图 10-33 所示为连接到指轮开关（TWS）的 BCD 输入接口模块。BCD 输入模块使处理器可以接受 4 位数字码。从这种接口输入数据到特定的寄存器或字存储器，供控制程序使用。寄存器输入模块一般接受直流 5V（TTL）到 24V 的信号。它们组合成一个包含了 16 或 32 个输入点的模块，分别对应了一个或两个 I/O 寄存器。数据处理指令用于接收从模块输入的信号，给用于控制程序中的寄存器提供数据。这样不需改动控制程序，就可以在外部改变给定值或设定值。



(a) 模块

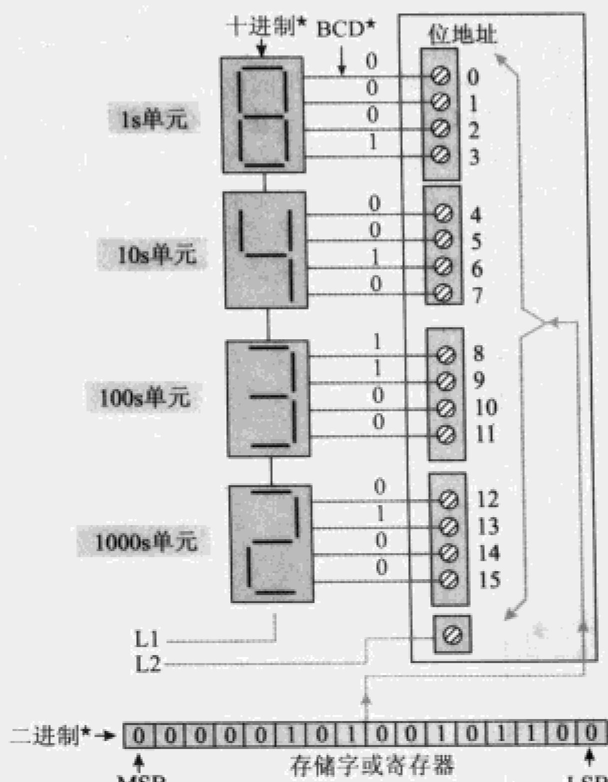


(b) 四位BCD指轮开关(获得Cincinnati Milacron公司许可)

图 10-33 BCD 输入模块

图 10-34 所示为连接了七段 LED 显示板的 BCD 输出接口模块。这种接口用于输出存储区中特定寄存器或字存储器中的数据。寄存器输出模块一般提供直流 5V（TTL）到直流 30V 的电压信号，并且有 16 或 32 个输出点，分别对应了一个或两个 I/O 寄存器。这种模块使 PLC 能对需要 BCD 码信号的设备进行操作。BCD 输出模块也可以用来驱动电流在 0.5A ~ 1A 范围内的小型直流负载。 288

图 10-35 所示为一段 PLC 程序，它使用了连接了指轮开关的 BCD 输入接口模块，和连接到 LED 显示板的 BCD 输出接口模块。LED 显示板监视指轮开关的十进制设置。程序中，指轮开关的设置与存储在等于指令 Source *B* 中的数字 100 进行比较，当输入开关 S1 闭合并且指轮开关的设定值等于 100 时，接通输出指示灯 PL。



(a) 模块

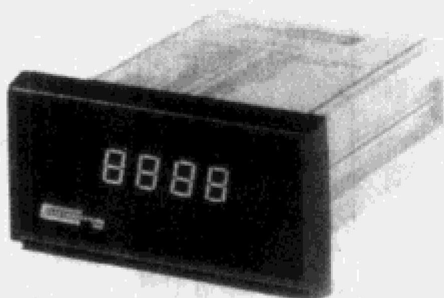

(b) BCD码转换为七段显示（获得Cincinnati Milacron公司许可）

图 10-34 BCD 输出接口模块

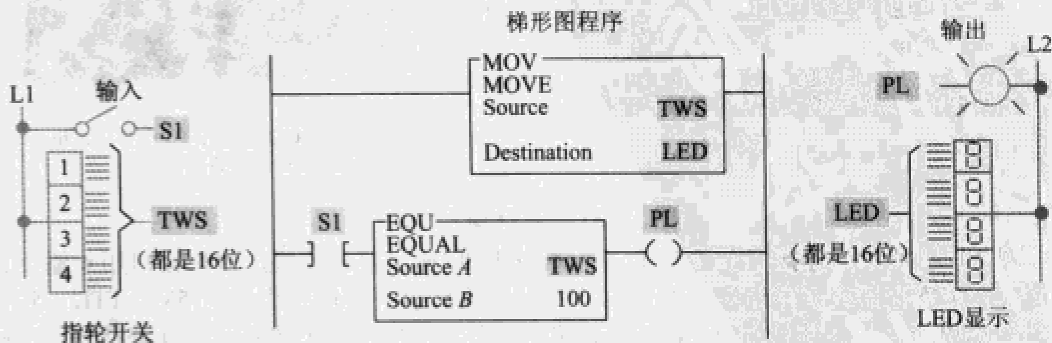


图 10-35 使用传送和等于指令的 BCD I/O 程序

输入模块和输出模块可以是位编址也可以是字编址，这使得在梯形图中编址更加方便。模拟量模块把模拟信号转换为 16 位数字信号（输入模块），或者把 16 位数字信号转化为模拟数据（输出模块）。例如，它们可以用于连接热电偶和压力变送器。

模拟量 I/O 允许对模拟量电压信号和电流信号进行监视和控制。图 10-36 显示了一个

模拟量输入模块的工作原理。模拟量输入模块中包含了合适的电路，能够接收现场设备的电压或电流信号。这些模拟电压或电流信号被 A/D 转换电路转换为数字量信号。转换后的数据与模拟信号成正比，并通过控制器的数据总线存储在特定的寄存器或字存储区中，以备接下来被控制程序使用。

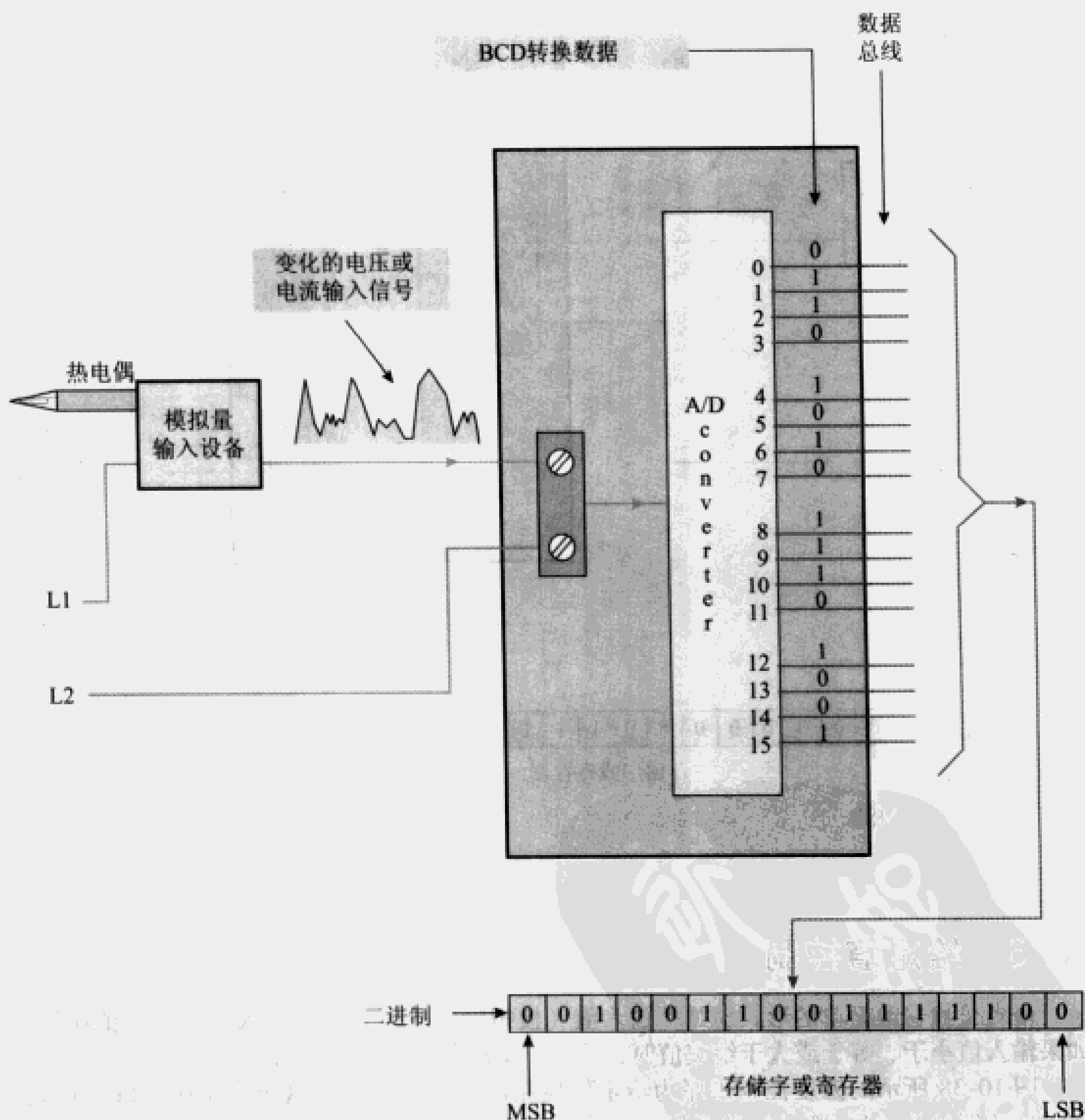


图 10-36 模拟量输入接口模块

模拟量输出接口模块接收来自处理器的数字信号，并把它转换为成比例的电压或电流信号，去控制现场的模拟量设备。图 10-37 所示为模拟量输出接口的工作原理。存储在特定寄存器或字存储区的数据，通过控制器的数据总线，传送到 D/A 转换器。从转换器输出的模拟量控制模拟输出设备。这些输出接口一般需要外加电源才能满足电流和电压要求。

289 ~ 291

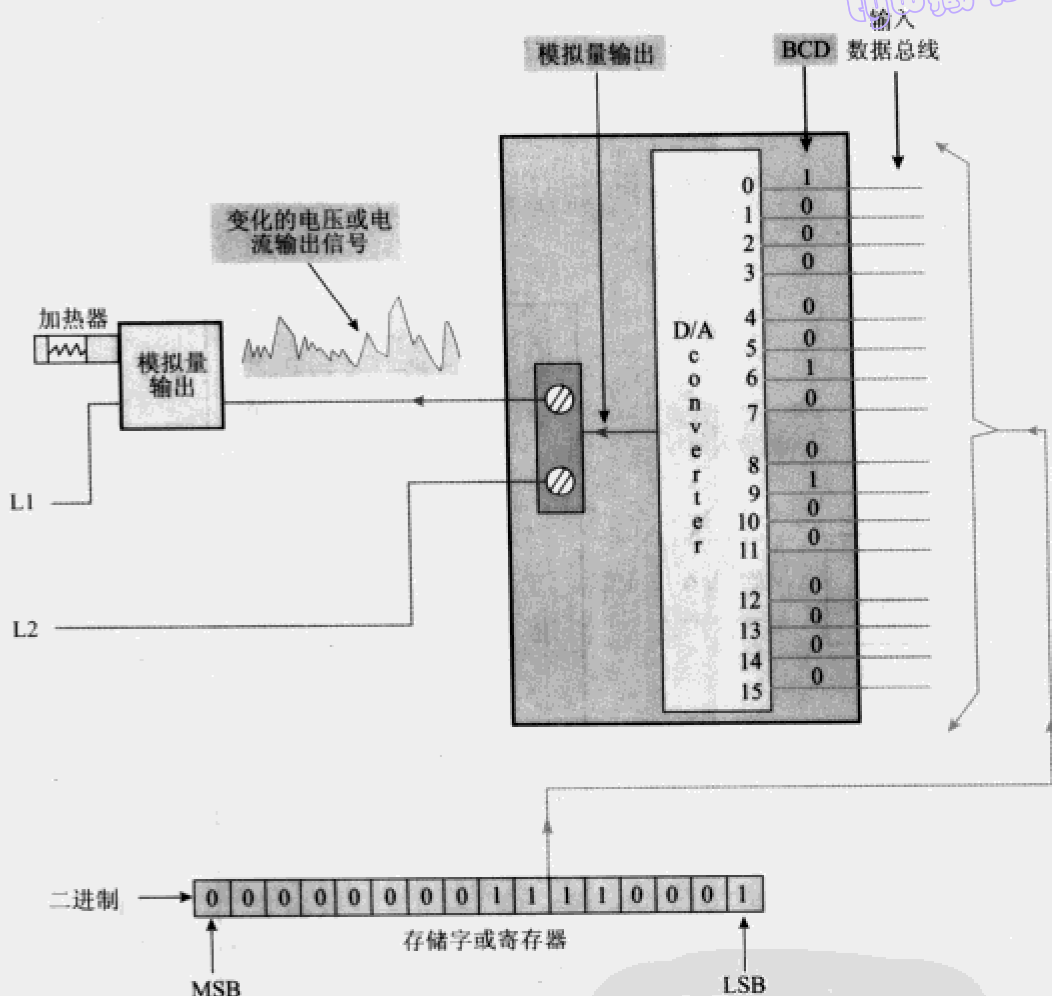


图 10-37　模拟量输出接口模块

10.6　给定值控制

给定值控制最简单的形式就是把输入值（模拟量或指轮开关的输入）与给定值比较。如果输入值小于、等于或大于给定值时，输出一个离散信号。

图 10-38 所示的温度控制程序为给定值控制的一个实例。在这个例子中，PLC 控制炉子加热器的简单通/断。炉温要保持在给定温度 600℉，允许 1% 的偏差。因此当炉子的温度下降到 597℉ 或更低时，加热器就要打开，并且一直保持到炉子的温度升高到 603℉ 或更高。加热器关闭直到温度下降到 597℉，一直往复循环。当小于等于指令为真，处于低温状态，程序控制加热器接通。当大于等于指令为真，处于高温状态，程序控制加热器关断。

不同的 PLC 模块可以实现几种常见的给定值控制形式。包括：通/断控制、比例（P）控制、比例－积分（PI）控制和比例－积分－微分（PID）控制。每一种都涉及使用某种形式的闭环控制，使像温度、压力、流量或液位这样的过程变量在保持期望的值。

292

典型的闭环控制系统框图如图 10-39 所示。测量值用于控过程变量，并与参考值或给

定值进行比较。如果实际值与期望值之间存在偏差，PLC 控制程序就会采取正确的调整，一直持续到期望值和实际输出之间的偏差在合理的范围内。

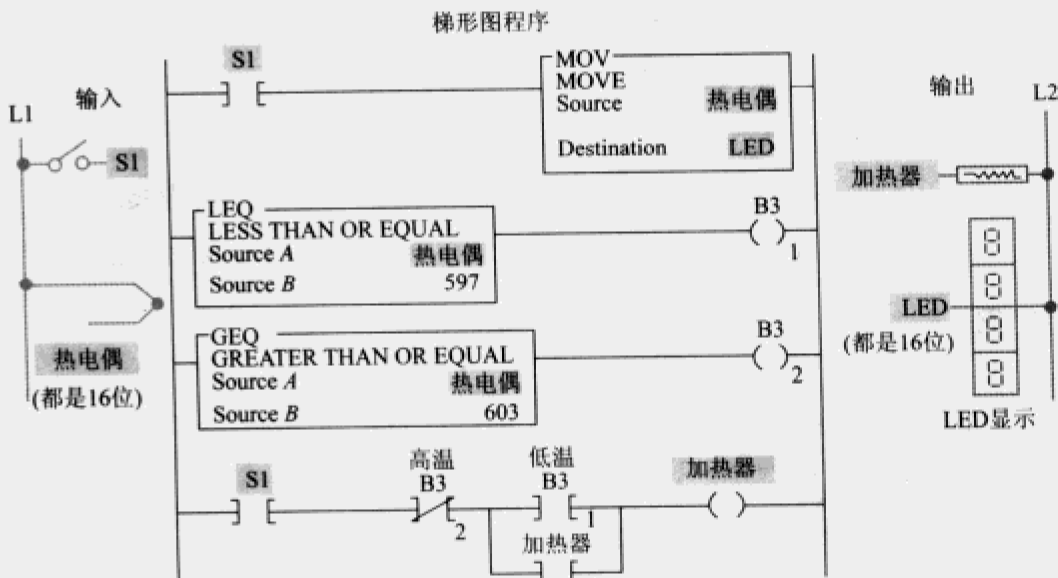


图 10-38 采用 Allen-Bradley PLC-5 或 SLC-500 指令的给定值温度控制

通/断控制（又称为双位置控制和继电器控制）的输出或末级控制元件的状态不是接通就是关断。当测量变量值大于给定值时关断，当测量变量值低于给定值时接通。控制器决不会使末级控制元件处在中间状态。

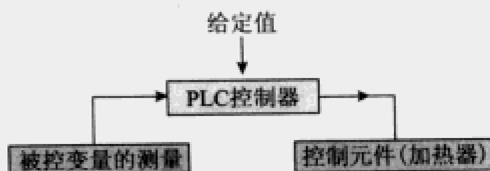


图 10-39 闭环控制系统框图

通/断控制对于许多过程和机器控制系统，成本低廉但是不够精确。通/断控制总会存在超调和系统往复。通常在给定值附近设置一个死区防止在给定值附近发生震荡。通/断控制不能调整特定系统的时间常数。

比例控制可以消除通/断控制的反复和不规则振荡，允许末级控制元件在接通和关断之间的中间位置。比例作用，根据测量变量值与期望值偏移量的大小，允许这个最终控制元件的模拟控制值来改变该处理器的能量值。 293

图 10-40 所示为一个比例控制过程的例子。PLC 的模拟量输出模块通过调整阀门打开的比例，控制流入存储槽的液体的多少。阀门开始打开 100%。随着液体液位接近给定值，处理器控制输出以不同的比例分级关闭阀门，保持液位在给定值。 294

比例－积分－微分（PID）控制是一种最复杂的并广泛应用的过程控制器。它基于数学分析，原理也很复杂，如图 10-41 所示。它根据系统偏差信号的幅值、累积和改变速率来阐述控制输出。系统突加的干扰，欲破坏系统原来的状态。与其他控制器相比，PID 控制器能更迅速的将系统偏差减少为 0。

PLC 要么安装了 PID I/O 模块，要么通过自身丰富的数学函数实现 PID 功能。比例部分的作用是产生正比于偏差信号的输出信号。这个偏差信号为测量信号与输入到 PLC 的给定信号的差值。积分控制作用是产生与预置偏差时间和大小成比例的输出信号。微分作用是产生与偏差改变速率成比例的输出信号。 295

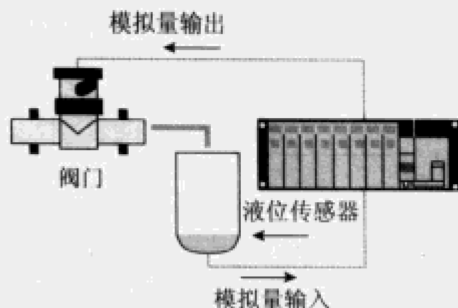


图 10-40 比例控制过程

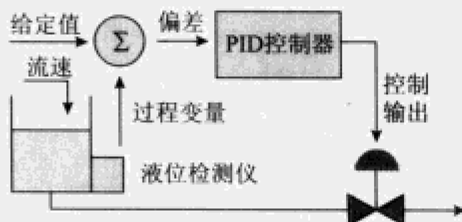


图 10-41 典型的 PID 控制回路

思考题

1. 解释寄存器（或字）和表（或文件）之间的差别。
2. PLC 的数据处理指令的功能是什么？
3. 数据处理指令可以分为哪两大类？
4. 数据传送指令包括哪些？
5. 解释图 10-42 中的逻辑梯级实现的功能？

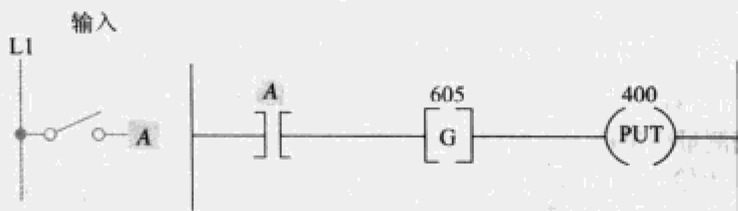


图 10-42

6. 用 MOV 指令把存储在字 N7:20 中的数据传送到字 N7:35，那么 Source 和 Destination 中分别要输入的地址是什么？
7. 解释屏蔽传送指令的功能。
8. 解释位域分配指令的功能。
9. 列写出三种类型的文件数据处理指令。
10. 列写出要输入到文件运算与逻辑指令（FAL）的六个参数和地址。
11. 如果 FAL 指令被设置为全模式，将会怎样影响数据传送？
12. 文件复制（COP）和文件填充（FLL）指令用于数据传送，相对于 FAL 指令的优势是什么？
13. 数据比较指令包含哪些？
14. 写出五种不同类型的数据比较指令的名称和符号。
15. 解释如何使用极限比较指令（LIM）对数据进行比较。
16. 多位 I/O 接口与离散 I/O 接口的区别？
17. 假设指轮开关设置成十进制数 3 286
    a. 等效的 BCD 数是多少？ b. 等效的二进制数是多少？
18. 假设一个热电偶连接到模拟量输入模块，解释热电偶的温度信号怎样与 PLC 的处理器通信。
19. 说出两种典型的模拟量输出现场设备的名称。
20. 借助原理框图，解释闭环控制系统的基本工作原理。
21. 解释图 10-43 中每个图的逻辑梯级实现的功能。

22. 比较末级执行元件的通/断控制和比例控制系统。
23. 解释由下面的 PID 模块的各个部分产生的输出信号类型。
    a. 比例控制　b. 积分控制　c. 微分控制

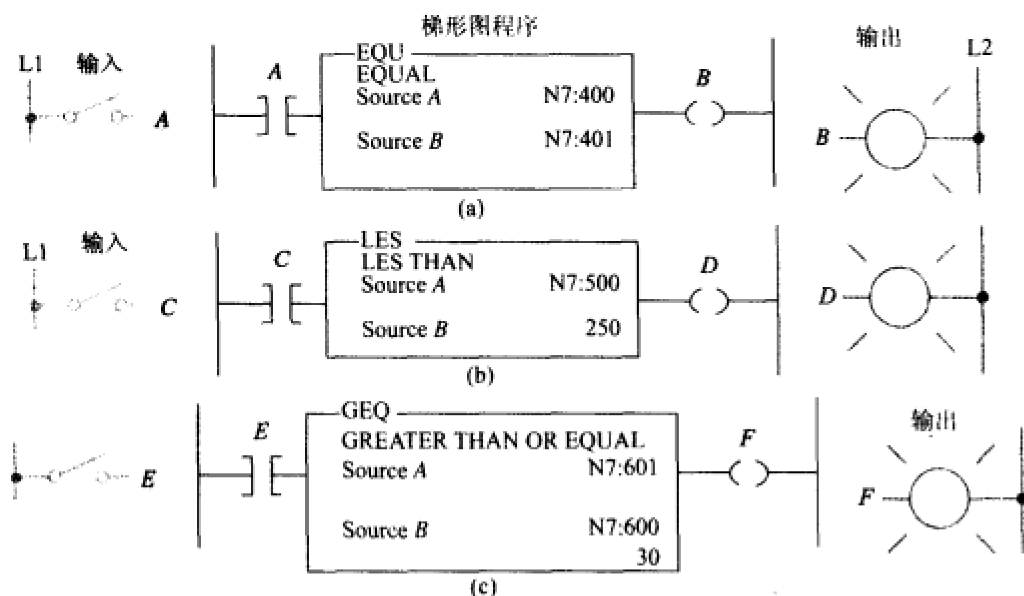


图　10-43

297

习题

1. 研究图 10-44 的数据传送程序，回答以下问题。
   a. 当 S1 断开，MOV 指令中整数字地址 N7:13 中存储的十进制数是多少？
   b. 当 S1 闭合，MOV 指令中整数字地址 N7:112 中存储的十进制数是多少？
   c. 当 S1 闭合，LED 显示的十进制数是多少？
   d. 如果要使 LED 显示十进制数 216，如何实现？

298

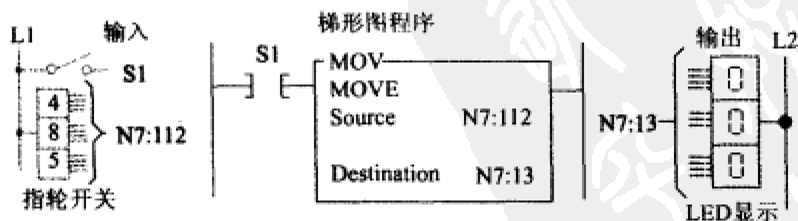
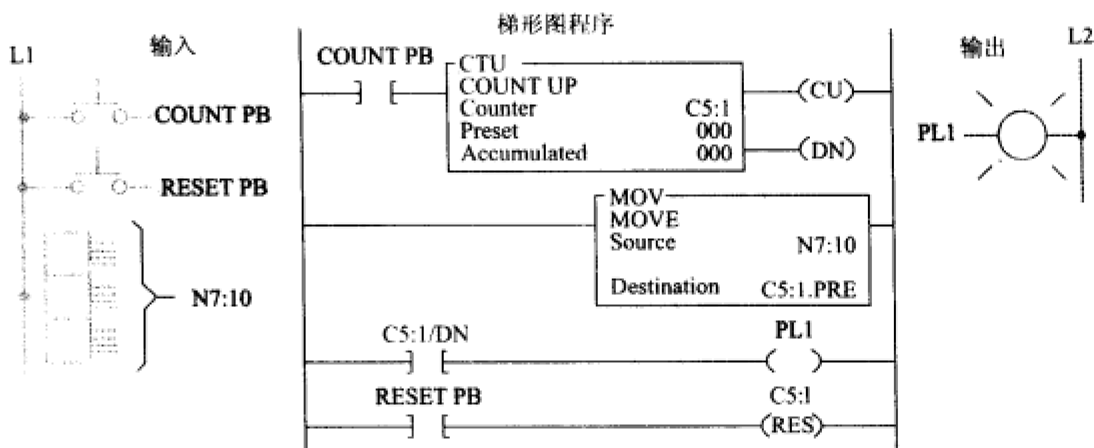


图　10-44

2. 研究图 10-45 的数据传送程序，回答以下问题。
   a. 什么决定了计数器中的预置值？
   b. 描述程序运行步骤，实现当输入 count PB 25 次上升沿后输出 PL1 接通。
3. 编写非保持型定时器程序，使延时一段时间后接通一个指示灯。使用指轮开关修改定时器的预置值。
4. 研究图 10-46 所示的数据比较程序，回答以下问题：
   a. 不管开关 S1 什么时候闭合，指示灯 PL1 都要亮吗？为什么？

b. 要改变存储在 EQU 指令中 Source *A* 中的数据，必须闭合 S1 吗？
c. 为了使输出指示灯变亮，指轮开关需要设置什么值？



299

图 10-45

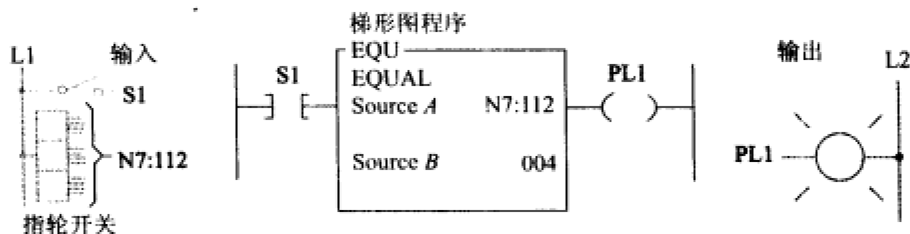


图 10-46

5. 研究图 10-47 所示的数据比较程序，回答以下问题：
   a. 写出使输出指示灯变亮的指轮开关的设定值。
   b. 如果开关 S1 打开，字 N7:112 中的数据为 003，指示灯会变亮吗？为什么？
   c. 假设 Source *B* 中存储的是一个加计数器的累加值，S1 闭合，当计数器累加值达到 150 时，指轮开关需要怎样设置才能使输出指示灯变亮？

300

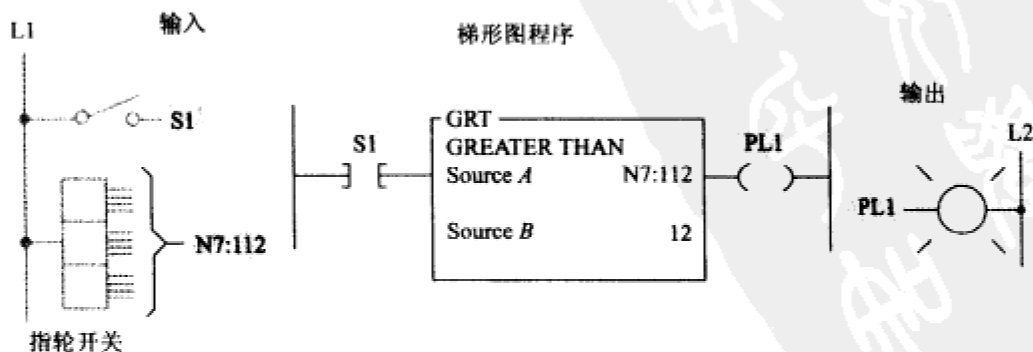


图 10-47

6. 编写程序实现下列功能
   a. 如果指轮开关的值小于 4 时接通指示灯 PL1

b. 如果指轮开关的值等于4时接通指示灯PL2
c. 如果指轮开关的值大于4时接通指示灯PL3
d. 如果指轮开关的值小于等于4时接通指示灯PL4
e. 如果指轮开关的值大于等于4时接通指示灯PL5

7. 编写程序实现把存储在地址N7:56中的数据复制到地址N7:60中。
8. 用屏蔽传送指令编写程序，实现把存储在地址I:2.0中的高8位数据传送到O:2.1中，屏蔽低8位。
9. 使用FAL指令编写程序，把从N7:40开始的整数数据文件中的20个字，复制到从N7:80开始的整数数据文件中。
10. 使用COP指令编写程序，实现把从B3:0开始的存储区中的128位数据，复制到从B3:8开始的存储区中。
11. 编写程序，实现当计数器的计数值在6到10之间时，灯变亮。
12. 编写程序，实现当计数器的计数值在小于10或大于30时，灯变亮。
13. 在4个小时内，每隔5分钟读取热电偶测得的温度值，并存储到特定的位置。每次读取的温度值存储到N7:150。最近4个小时的数据存储到文件#7:200。

第 11 章　数 学 指 令

学习目标：

- ❑ 分析并解释应用于 PLC 程序的数学指令。
- ❑ 编写包含数学指令的 PLC 程序。
- ❑ 在控制过程中应用 PLC 的算术运算功能。

大多数 PLC 都具有算术运算能力，当需要时 PLC 每秒可以进行许多运算操作来进行快速更新时间。一般的 PLC 运算功能更新间隔为 1 个或 2 个扫描周期。本章包含了 PLC 及其应用执行的基本数学功能。

302

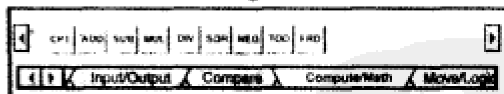
11.1　数学指令

数学指令，如同数据处理指令一样，使 PLC 能够实现计算机系统的一些功能。PLC 的数学处理能力不是让它去代替一个计算器，而是对存储在存储字中的数据进行算术运算操作。例如，假定用计数器跟踪生产部件的个数，想要显示达到一定量还需生产的部件数，要显示这个数据，就需要用期望值减去计数器中的累加值。还有其他的一些应用包括：联合部分计数，减去检测到的亏量和计算运行速度等。

根据使用的 PLC 的处理器类型，许多不同的数学指令能够由编程实现。PLC 能实现的基本数学功能包括：加（+）、减（-）、乘（×）和除（÷）。这些指令对存储在两个字或寄存器中的数据执行期望的功能。PLC 的数据处理指令（数据传送和数据比较）被用来用数学符号执行数学功能。数学指令全都是输出指令，图 11-1 所示为基于 Allen-Bradley SLC-500PLC 及其相关的软件 RSLogix 的典型数学指令。

303
|
304

软件RSLogix的一部分



命　　令	名　　称	解　　释
CPT	计算指令	计算一个表达式并将计算结果存储在目的地址中
ADD	加法指令	把 Source A 中和 Source B 中的数据相加，并将计算结果存储到目的地址
SUB	减法指令	把 Source A 中的数据减去 Source B 中的数据，并将计算结果存储到目的地址
MUL	乘法指令	把 Source A 中和 Source B 中的数据相乘，并将计算结果存储到目的地址
DIV	除法指令	把 Source A 中的数据除以 Source B 中的数据，并将计算结果存储到目的地址
SQR	平方根指令	计算 Source 地址数据的平方根，并把结果的整数存放到目的地址
NEG	取反指令	改变 Source 值的符号，并把结果存放到目的地址中
TOD	整数转换为 BCD 码	把 16 位整数转换为 BCD 码存放于寄存器或目的地址中
FRD	从 BCD 码转换为整数	把存储在数学寄存器或 Source 地址中的 BCD 码转换为整数并将结果存放于目的地址

图 11-1　基于 Allen-Bradley SLC-500PLC 及其相关软件 RSLogix 的数学指令

11.2 加法指令

加法指令把存储在相关存储区中的两个数据相加，如何访问这些数据取决于控制器。

Allen-Bradley PLC-5 和 SLC-500 控制器采用模块式的加法（ADD）指令执行加功能。图 11-2 为加法指令的例子。在这个例子中，当梯级为真时，存储在 Source *A* 地址 N7:0 中的值 25 与存储在 Source *B* 地址 N7:1 中的数据 50 相加，结果 75 被存放于目的地址 N7:2 中。Source *A* 和 Source *B* 中可以是数据也可以是包含了数据的地址，但是两个不能全为常数。

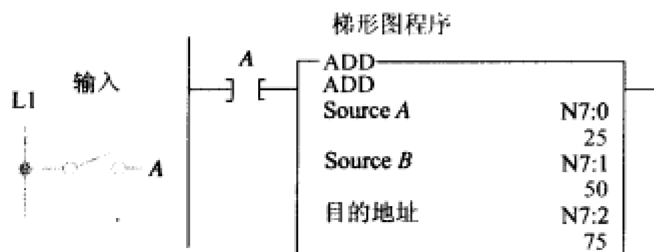


图 11-2 Allen-Bradley PLC-5 和 SLC-500 的加法指令（ADD）

图 11-3 所示程序为用 ADD 指令把两个加计数器的累加计数值相加。在这个应用中，当两个计数器的计数值之和大于等于 350 时，指示灯亮。ADD 指令中的 Source *A* 中存储了计数器 C5:0 的累加值，Source *B* 中存储了计数器 C5:1 的累加值。Source *A* 的值加上 Source *B* 的值，结果存储在目的地址 N7:1 中。大于等于指令的 Source *A* 中存储了目的地址 N7:1，Source *B* 中存储了常数 350。因此当两个计数器的累加值之和大于等于 350 时，大于等于指令为真。复位按钮用来把两个计数器的累加值复位为 0。

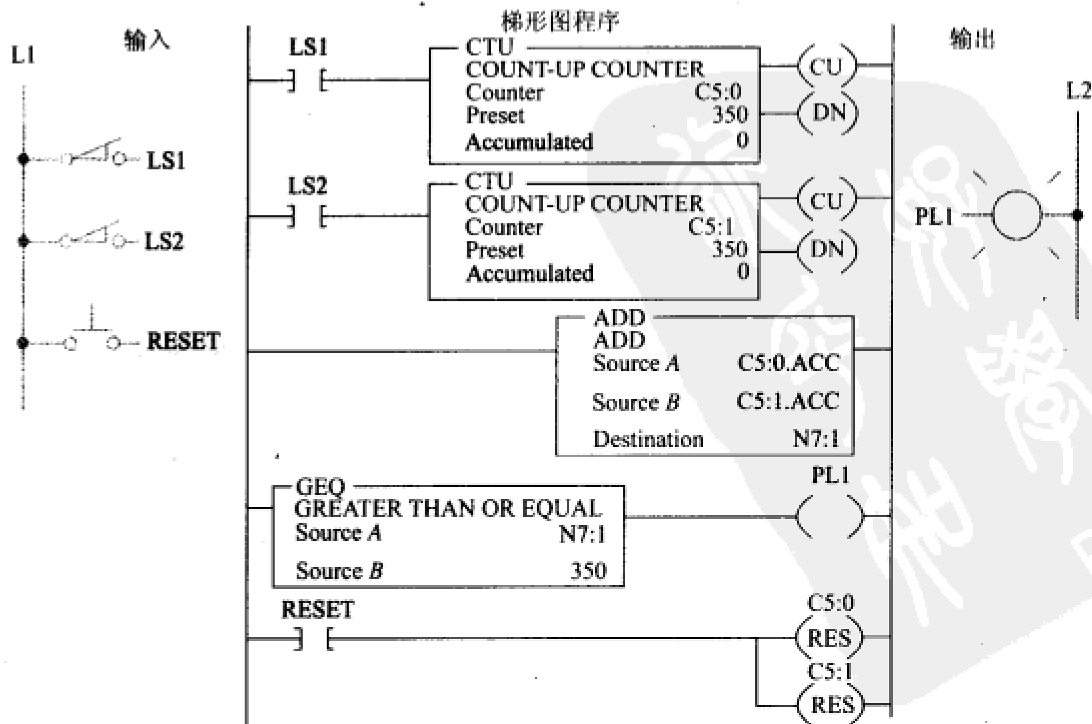


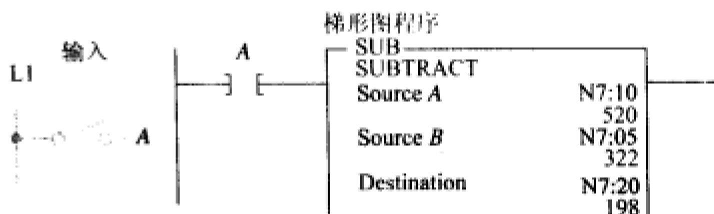
图 11-3 使用 ADD 指令的 PLC-5 和 SLC-500 计数器程序

当使用数学函数时，要注意确保数据在数据表和文件存储的范围内，否则溢出位被置位。例如 Allen-Bradley PLC-5 控制器，不能在整数地址中存储超过 32 767 的数据。

11.3 减法指令

减法（SUB）指令是一个输出指令，它用一个数据减去另一个数据，把结果存储到目的地址中。当梯级条件为真，减法指令把 Source *A* 中的数据减去 Source *B* 中的数据，将结果存储到目的地址中。

[305] 图 11-4 所示为模块式减法（SUB）指令的一个例子，SUB 指令为 Allen-Bradley PLC-5 和 SLC-500 控制器指令设置的部分。在这个例子中，当梯级为真时，存储在 Source *A* 地址 N7:10 中的数据 520 减去 Source *B* 地址 N7:05 中的数据 322，将结果 198 存储到目的地址 N7:20 中。Source *A* 和 Source *B* 中可以是数据或者是包含了数据的地址。但是两个不能都是常数。



[306] 图 11-4 Allen-Bradley PLC 模块式的减法指令（SUB）

图 11-5 所示的程序是显示如何用减法（SUB）指令监视容器填充溢出状态。在这个应用中，需要一个报警器，当达到设定重量 500 磅，填充系统又有 5 磅或更多的材料渗漏到容器中时，发出报警。当按下开始按钮，梯级 1 的填充电磁阀接通，梯级 2 的监视指示灯变亮，材料开始被填充进容器。梯级 3 的程序对容器填充时的重量进行连续监视。当重量达到 500 磅时，填充电磁阀被关断，填充停止。同时填充指示灯也熄灭，梯级 3 的充满指示灯变亮。这时，有 5 磅或更多原材料漏到容器中，梯级 5 中的报警器报警，并保持到溢出的重量减小到 5 磅范围以内。

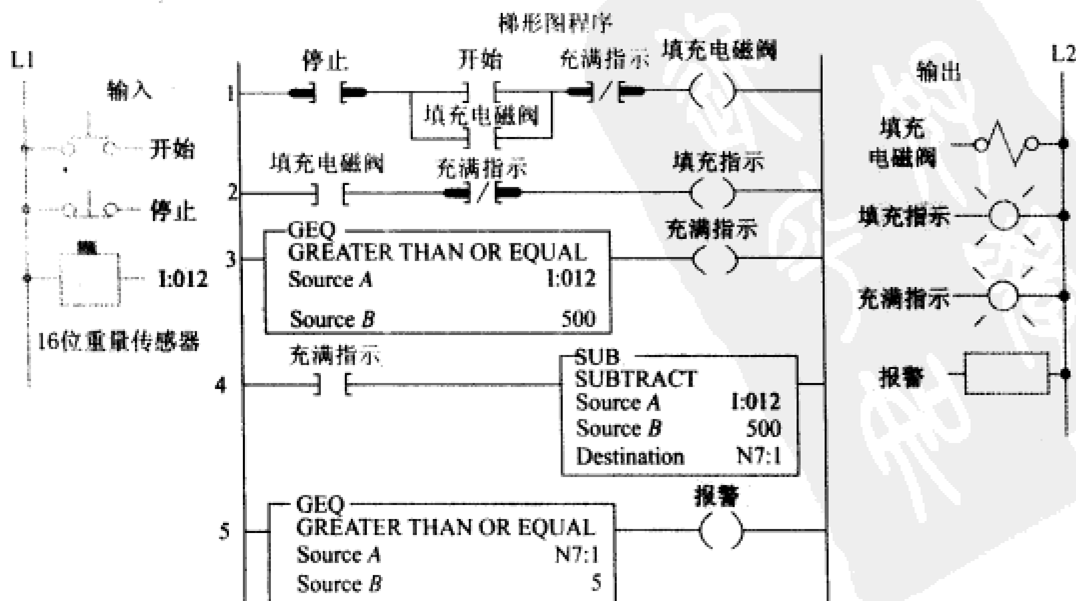


图 11-5 Allen-Bradley PLC-5 和 SLC-500 的溢出报警程序

11.4 乘法指令

乘法（MUL）指令是一个输出指令，两个数相乘结果存储到目的地址。图 11-6 为一个 MUL 指令的例子。在这个例子中，存储在 Source *A* 中的常数 20 乘以存储在 Source *B* 中的计数器 C5:10 的累加值，结果被存放在目的地址 N7:2 中。和前面的数学指令一样，乘法指令中 Source *A* 和 Source *B* 中可以是值（常数）也可以是地址，但不能全部为常数。 [307]

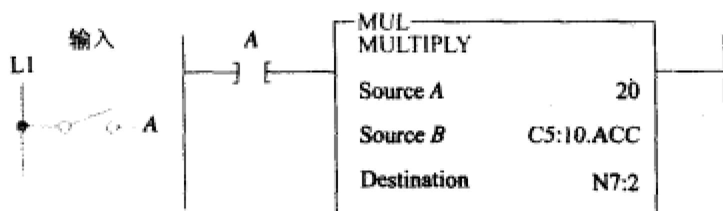


图 11-6 乘法（MUL）指令

图 11-7 为一个简单的乘法（MUL）程序。当输入 *A* 接通，存储在 Source *A* 地址N7:1 中的 123，与存储在 Source *B* 地址 N7:2 中的数据 61 相乘，乘积结果 7 503 被放入目的地址字 N7:3 中。因此，等于指令变为真，输出指示灯 PL1 亮。

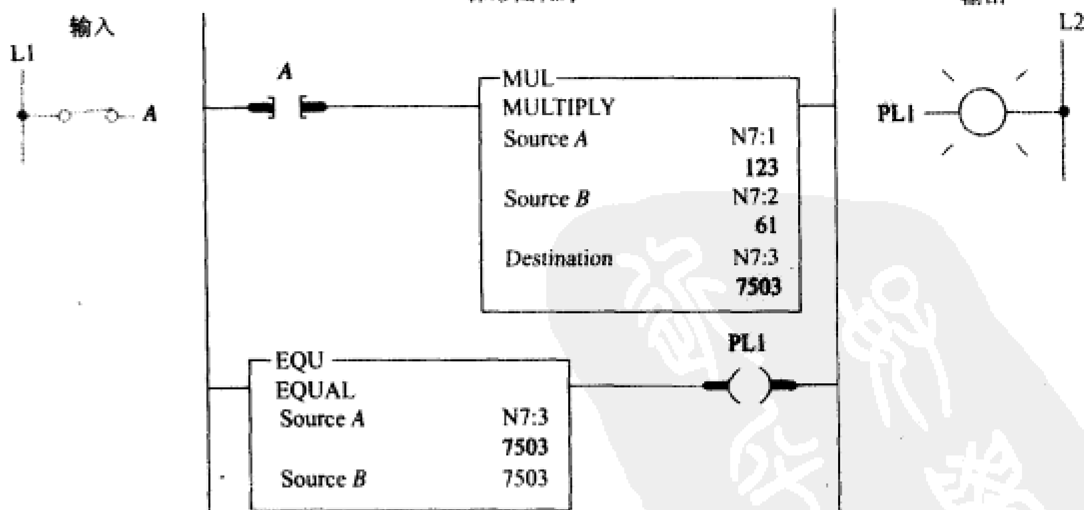


图 11-7 简单的乘法程序

图 11-8 为用 Allen-Bradley PLC-5 和 SLC-500 乘法指令实现的炉温控制程序。在这个程序中，PLC 计算设定值的高、低死区或通/断限制。不论设定值为多大，高、低限制自动设为设定值的 ±1%。温度的给定值通过指轮开关调整。一个模拟量热电偶接口模块用于监视当前的炉温。因此当炉温下降到 396℉ 电子加热器接通，并保持直到温度上升到 404℉ 以上。如果温度给定值改为 100℉，死区允许范围为仍旧是 ±1%，即下限为99℉，上限为 101℉。存储在字 N7:1 中的数据代表温度上限，存储在字 N7:2 中的数据代表温度下限。 [308]

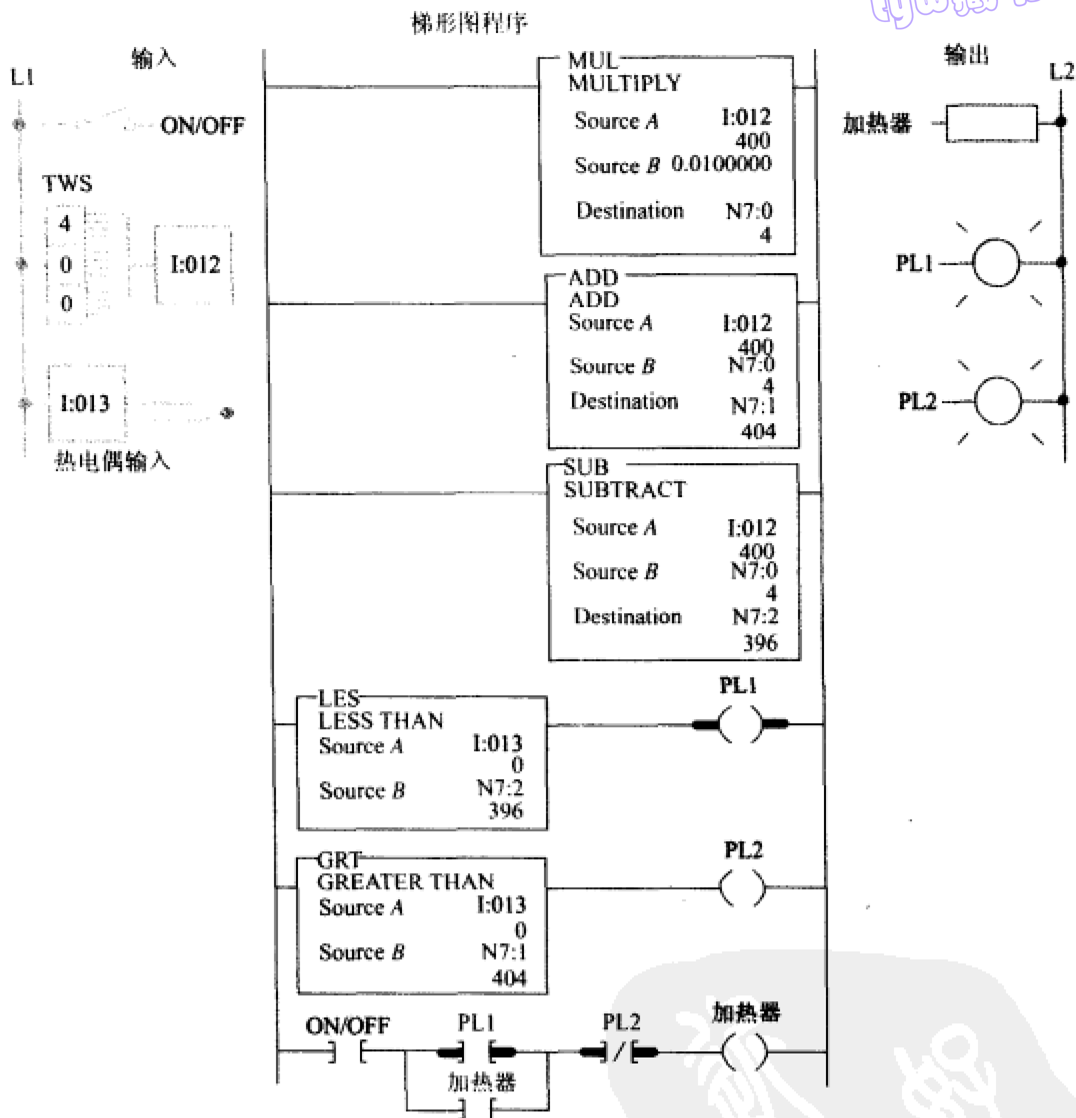


图 11-8　炉温控制程序

11.5　除法指令

除法（DIV）指令使存储在 Source *A* 中的数据被存储在 Source *B* 中的数据除，结果存储到目的地址和算术寄存器。如果余数大于等于 0.5，则要向上取整。不被整除的商被放 [309] 于算术寄存器的高位字节，余数被放置于低位字节。有些大型的 PLC 支持浮点数，和整数一样。例如，10 被 3 除的结果为 3.333 333（浮点数）或者是商为 3 余数为 1。图11-9为 DIV 指令的一个例子。在这个例子里，Source *A* 中的数据计数器 C5:10 的累加值被 Source *B* 中的常数 2 除，结果放置于目的地址 N7:3。

图 11-10 为一个简单的 DIV 程序。当输入 *A* 接通时，存储在 Source *A* 地址 N7:0 中的 120，被存储在 Source *B* 地址 N7:1 中的 4 除，结果 30 存储到目的地址 N7:5，结果等于指令变为真，输出指示灯 PL1 变亮。



图 11-9 除法（DIV）指令

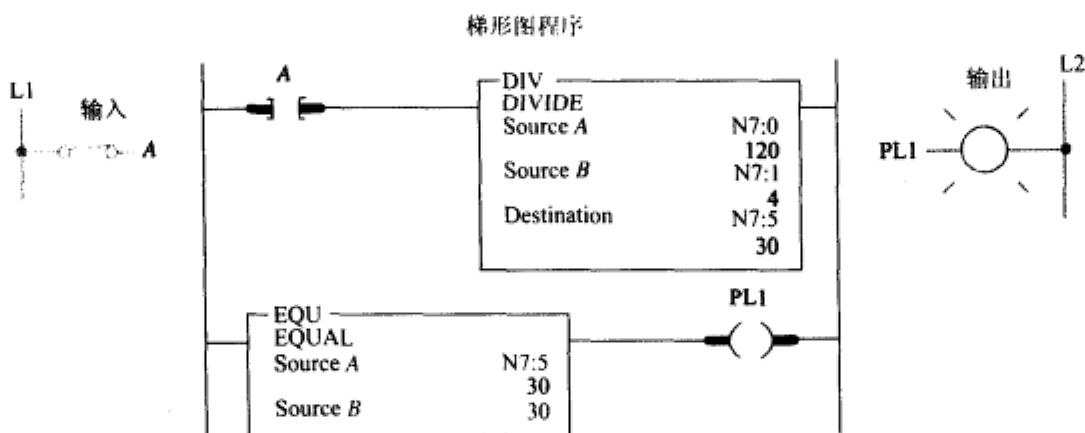


图 11-10 简单的除法程序

图 11-11 所示为如何用 DIV 指令作为程序的一部分把摄氏温度转换为华氏温度。在这个应用中，连接到输入模块的指轮开关指示输入摄氏温度值。程序将数据表中的摄氏温度值转换为华氏温度值并显示。

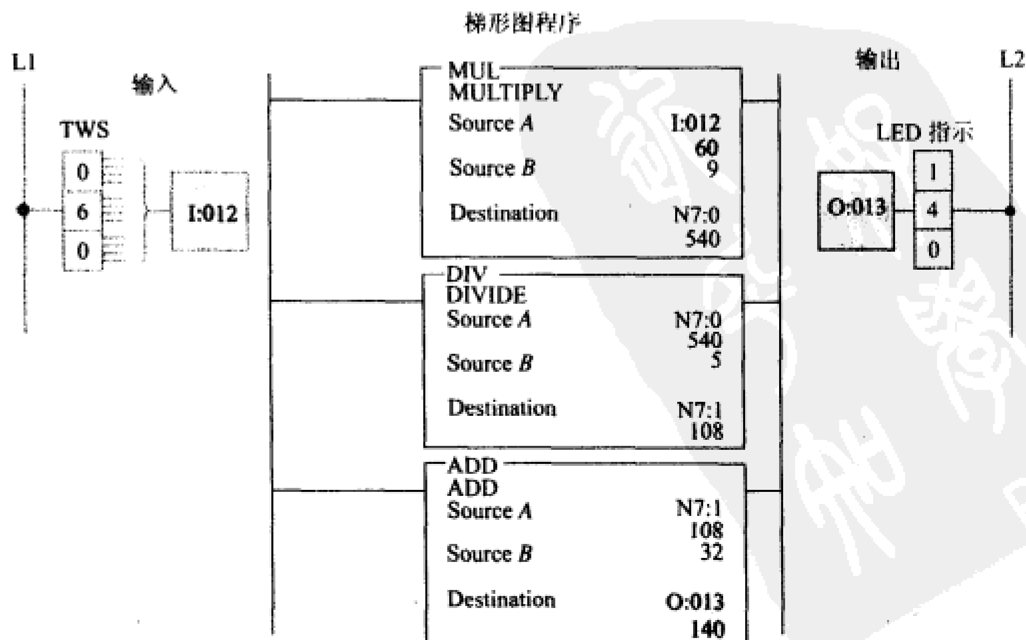


图 11-11 采用 Allen-Bradley PLC-5 或 SLC-500 指令转换摄氏温度值为华氏温度值

公式

310

$$°F = \left(\frac{9}{5} \cdot °C\right) + 32$$

为程序的基础。在这个例子中，假设读取当前温度值为60℃，乘法指令将60℃乘以9，存储乘积540到地址N7:0中。接下来DIV指令把540除以5，将结果108存储到地址N7:1。最后，ADD指令加32到108上，和140存储到地址O:13中，因此60℃ = 140°F。

11.6 其他的字数学指令

311

图11-12所示为平方根（SQR）指令的一个例子，待求平方根的数据放在Source地址中。当指令为真，计算数据的平方根，并把结果放置于目的地址。如果Source地址中的数据为负数，就把其绝对值的平方根存储到目的地址。

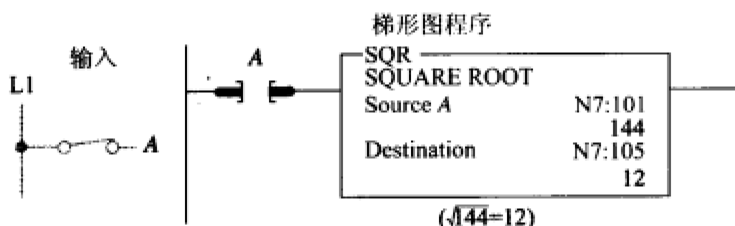


图 11-12 平方根指令（SQR）

图11-13为取反（NEG）指令的一个例子，当指令为真时，指令把存储在Source地址N7:52中的数据取反（改变符号），存储结果到目的地址N7:53中。正数直接以二进制的形式存储，负数采用第二种补码形式存储。



图 11-13 取反指令（NEG）

清零（CLR）指令用于设定目的地址字中的数据为0。图11-14的例子中，当指令为真时，目的地址N7:22中的数据被清零。



图 11-14 清零指令（CLR）

整数转换为BCD码（TOD）指令用于把16位整数转换为二进制形式的十进制数（BCD码）。这个指令用于把以二进制形式存储在处理器中的数据，传送到具有BCD码形式的外部设备，如LED显示板。图11-15所示的例子，将存储在Source地址N7:23中的二进制形式的数据，转换为相同大小的十进制BCD码数据，存储到目的地址O:20中。Source地址中数据为十进制10，但是目的地址中显示的是16。由于处理器都是以二进制

形式存储数据的，16 就是二进制形式的 BCD 码。二进制形式 10 的 BCD 码和二进制形式的 16 相同。

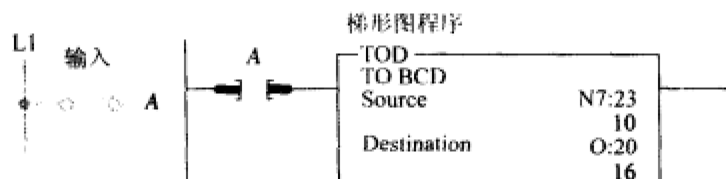


图 11-15 整数转换为 BCD 码（TOD）指令

BCD 码转换（FRD）指令用于把二进制码的十进制（BCD）数据转换为整数。这个指令用于将外部 BCD 设备（如 BCD 指轮开关）提供的数据，转换为处理器能够操作的二进制数。在图 11-16 所示的例子中，把存储在源地址 I:30 中的 BCD 码数据转换为二进制形式的大小相等的十进制数，并存储到目的地址 N7:24 中。

数据整定（SCL）指令用于非常大或非常小的数据，通过比率值把它变大或变小。当梯级条件为真，通过指令使源地址中的数据与特定比率值相乘，取整后加上偏移量，结果放置于目的地址。可以用此指令对来自模拟量模块的数据整定，把它变为各种处理器或其他的模拟量模块规定的数据范围内。例如，可以用 SCL 指令把 4 - 20mA 的输入信号转换为 PID 过程变量值，或者整定一个模拟量输入控制一个模拟量输出。在图 11-17 的例子中，存储在源地址 N7:0 中的 100 与 25 000 相乘，然后除以 10 000，最后加上 127，结果 377 放置于目的地址 N7:1 中。 312

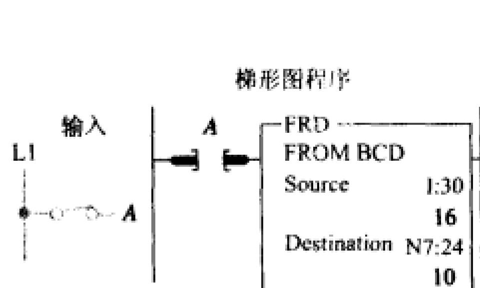


图 11-16 BCD 码转换（FRD）指令

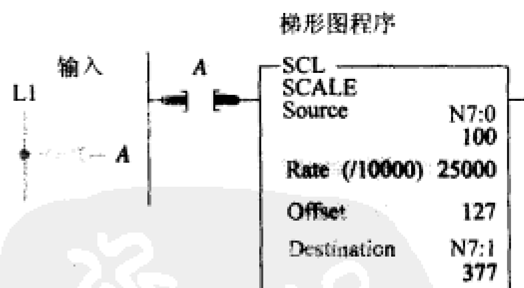


图 11-17 数据整定（SCL）指令

313

11.7 文件算术运算

文件算术功能包括：文件加、文件减、文件乘、文件除、文件平方根、文件从 BCD 码转换和文件转换为 BCD 码指令。文件算术与逻辑（FAL）指令可以结合文件传送指令一起使用。FAL 指令执行的算术运算操作为 ADD、SUB、MULT、DIV 和 SQR。

FAL 指令的文件加法功能用于执行多个字的加操作。在图 11-18 的例子中，当梯级变为真时，表达式指示处理器将文件地址 N7:25 中的数据与文件地址 N7:50 中数据相加，结果存储到文件地址 N7:100 中。每个扫描周期的模式选择全模式，因此在一个扫描周期内完成所有的操作。

图 11-19 所示为 FAL 指令文件减法功能的一个例子。在这个例子中，当梯级为真时，处理器从文件地址 N10:0 中的每一个字都减去一个常数（255），结果存储到目的文件地址 N7:255 中。模式设置为 2，因此从指令为真开始，到完成整个操作需要两个扫描周期。 314

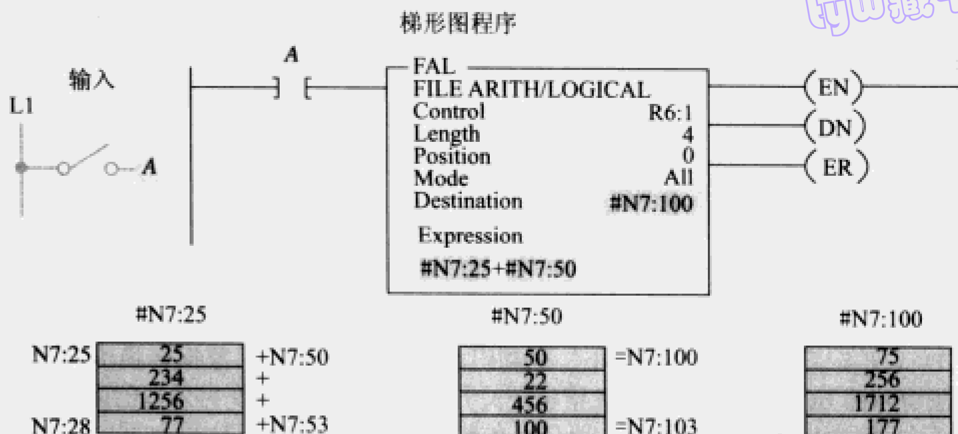


图 11-18　FAL 指令的文件加法功能

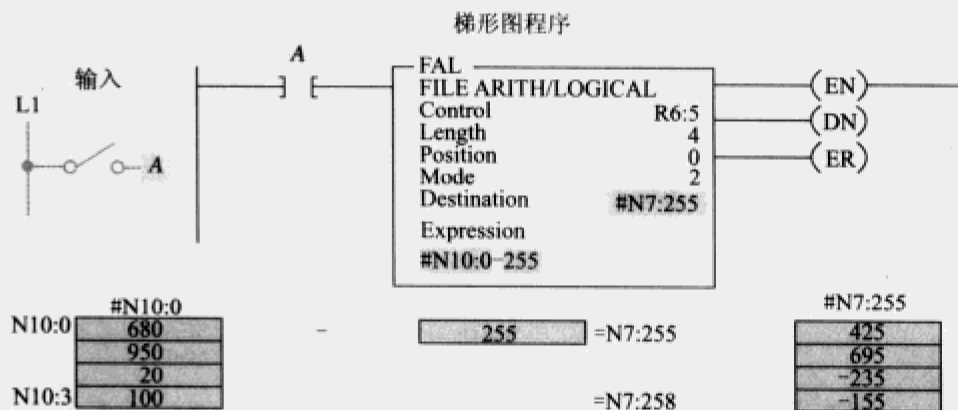


图 11-19　FAL 指令的文件减法功能

图 11-20 为 FAL 指令文件乘法功能的一个例子，在这个例子中，当梯级变为真，在文件地址 N7:330 中的数据与元件地址 N7:23 中的数据相乘，结果存放于文件地址 N7:500 中。

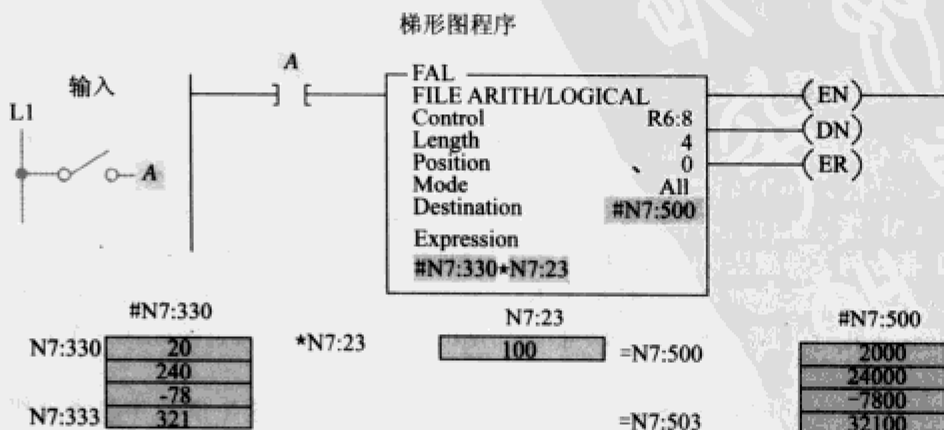


图 11-20　FAL 指令的文件乘法功能

图 11-21 为 FAL 指令文件除法功能的一个例子，在这个例子中文件地址 F8:20 中的数据被文件地址 F8:100 中的数据除，结果存储到单元地址 F8:200 中。模式为增量式，因此当指令的每次上升沿到来时，指令对一组元件进行操作。

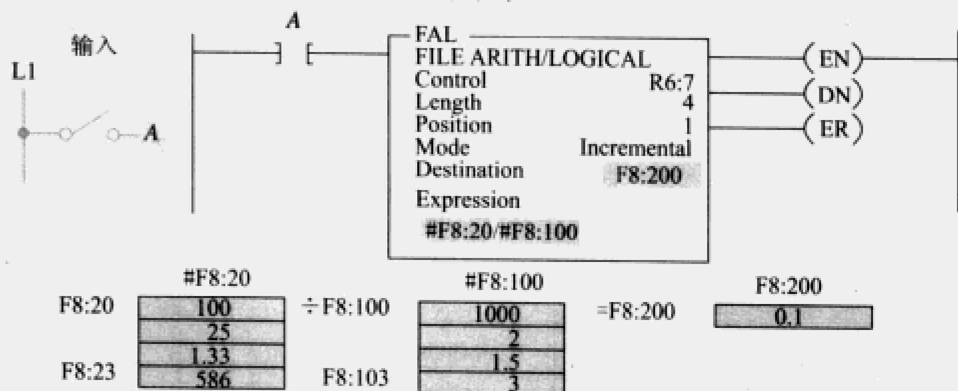


图 11-21 FAL 指令的文件除法指令

思考题

1. 解释 PLC 应用数学指令的目的。
2. 一些 PLC 中的四种基本的数学功能是什么？
3. PLC 数学指令的标准形式是什么？
4. 数学指令属于输入指令还是输出指令？
5. 参考图 11-22，当 N7:3 中的数据为 60，N7:20 中的数据为 80，那么 Source B 中的数据是多少？
6. 参考图 11-23，如果 N7:3 中的数据为 500，那么目的地址（Destination）中的数据是多少？

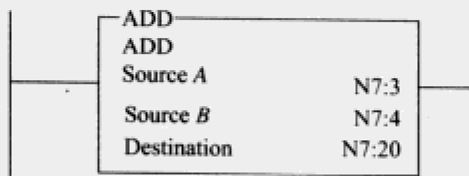


图 11-22

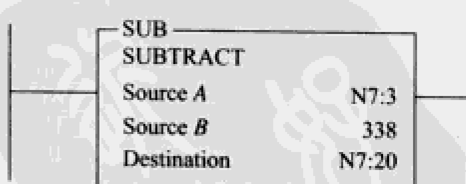


图 11-23

7. 参考图 11-24，如果 N7:3 中的数据为 40，N7:4 中的数据为 3，那么目的地址中的数据是多少？
8. 参考图 11-25，如果 N7:3 中的数据为 15，N7:4 中的数据为 4，那么目的地址中的数据是多少？

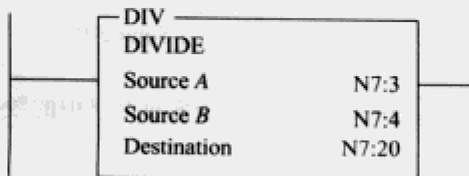


图 11-24

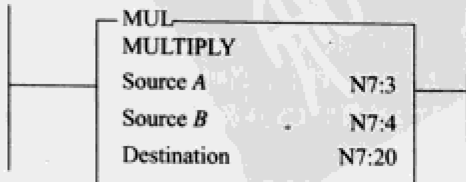


图 11-25

[316] 9. 参考图11-26，如果N7:3中的数据为-345，那么N7:20中的数据是多少？

10. 参考图11-27，当梯级变为真时，字B3:3中的每一位的值分别是多少？

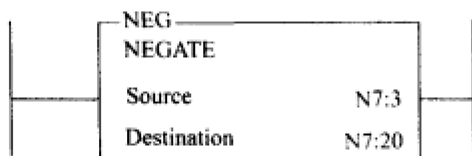


图 11-26

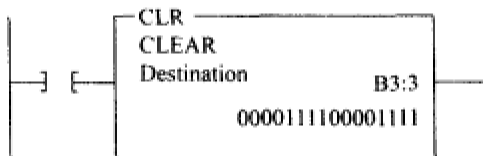


图 11-27

11. 参考图11-28，当梯级变为真时，N7:101中的数据是多少？

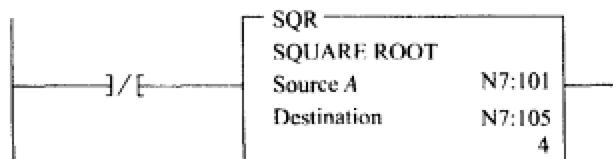


图 11-28

[317] 12. 参考图11-29，当梯级变为真时，文件#7:10中存储的数据是多少？

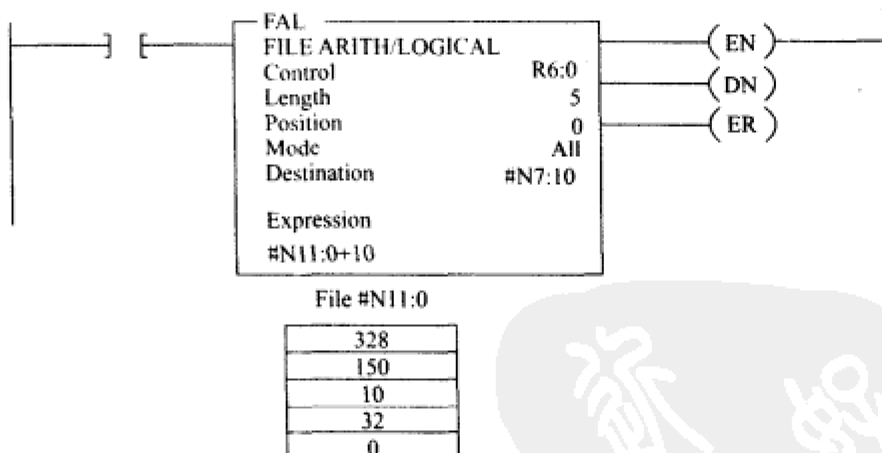


图 11-29

习题

1. 参考图11-3的计数器程序回答下列问题：

   a. 假设计数器C5:0和C5:1的累加计数值分别为148和36，分别说出下面每个字中的数据是多少？

      （1）C5:0. ACC （2）C5:1. ACC （3）N7:1 （4）GEQ指令的Source B

   b. 此时输出PL1变亮吗？为什么？

   c. 假设计数器C5:0和C5:1的累加计数值分别为250和175，说出下面每个字中的数据是多少？

      （1）C5:0. ACC （2）C5:1. ACC （3）N7:1 （4）GEQ指令的Source B

   d. 此时输出PL1变亮吗？为什么？

2. 参考图 11-5 的填充报警程序，回答下面的问题：
   a. 假设容器正在填充，刚好达到 300 磅，说出此时每个逻辑梯级的状态（真或假）。
   b. 假设容器正在填充，刚好达到 480 磅，说出存储在下面每一个字中的数据是多少？
      （1）I:012 （2）N7:1
   c. 假设容器填充到 502 磅，说出在这种条件下每个逻辑梯级的状态（真或假）。
   d. 假设容器填充到 510 磅，说出在这种条件下存储在下面每一个字中的数据是多少？
      （1）I:012 （2）N7:1
   e. 容器填充到 510 磅时，说出每个逻辑梯级的状态（真或假）。 318
3. 参考图 11-8 的炉温控制程序，回答下列问题：
   a. 假设温度给定值为 600℉，电子加热器接通和关断的温度值是多少？
   b. 假设温度给定值为 600℉，热电偶输入模块检测到的温度值 590℉，此时下面每个字中的数据分别是多少？
      （1）I:012 （2）I:013 （3）N7:0 （4）N7:1 （5）N7:2
   c. 假设温度给定值为 600℉，热电偶输入模块检测到的温度值为 608℉，此时下面每个输出的状态是什么（接通或关断）？
      （1）PL1 （2）PL2 （3）加热器（heater）
4. 参考图 11-11 的摄氏温度转换为华氏温度的程序，说出当指轮开关设定为 035 时，下面每个字中存储的数据分别是多少？
   （1）I:012 （2）N7:0 （3）N7:1 （4）O:013
5. 设计一段程序，只要当输入 *A* 接通时，就把存储在 N7:23 和 N7:24 中的数据相加，结果存储到 N7:30 中，然后当输入 *B* 为真时，把 N7:30 中的数据复制到 N7:31 中。
6. 设计一段程序，获取通延时定时器（TON）T4:1 的累加值，并把它以 4 位 BCD 码形式显示到 LED 显示板上。连接 LED 的地址为 O:023。当输入 *A* 为真时，通过一组 4 位的 BCD 码指轮开关改变定时器的预置值。连接指轮开关的地址为 I:012。
7. 设计程序实现下面的算术功能：
   a. 使用 MOV 指令传送 45 到 N7:0，传送 286 到 N7:1。
   b. 把上面两个数据相加，结果存储到 N7:2。
   c. 把 N7:2 中的数据减去 785，存储结果到 N7:3。
   d. 把 N7:3 中的数据与 25 相乘，结果存储到 N7:4。
   e. N7:4 中数据除以 35，存储结果到 F8:0。 319
8. a. 通过三条部件传送带生产线（1-2-3）向一条主传送带送料。这三个传送带生产线都分别带有计数器。设计一段 PLC 程序，获得主传送带上面的部件总数。
   b. 给程序中增加一个定时器，每隔 30s 更新一次部件总数。
9. 参考图 11-30，当输入为真时，下面每个字中存储的数据是多少？
   a. N7:3 b. N7:5 c. F8:1
10. 参考图 11-31，当输入为真时，下面每个字中存储的数据是多少？ 320
    a. N7:3 b. N7:4 c. N7:5 d. N7:6
11. A、B 两条传送带生产线给主传送带 M 送料，靠近主传送带的第三条传送带 R 用来移走主传送带 M 上的报废产品，A、B 和 R 上面都带有部件计数器。设计一段 PLC 程序，获得主传送带 M 上面的部件总数。
12. A、B 两条传送带生产线给主传送带送料，A 向主传送带传送装有 6 包每箱的苏打，B 向主传送带传送装有 8 包每箱的苏打，A、B 传送带都带有对箱子进行计数的计数器。设计一段 PLC 程序，计算出在主传送带上面的苏打总包数。 321

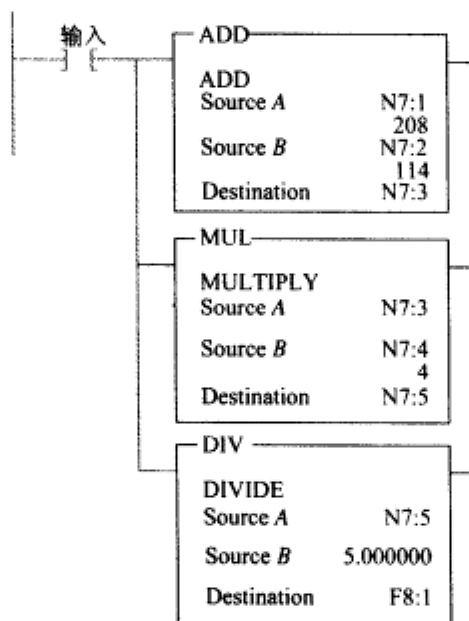

输入
ADD
ADD
Source A N7:1
208
Source B N7:2
114
Destination N7:3
MUL
MULTIPLY
Source A N7:3
Source B N7:4
4
Destination N7:5
DIV
DIVIDE
Source A N7:5
Source B 5.000000
Destination F8:1

图 11-30

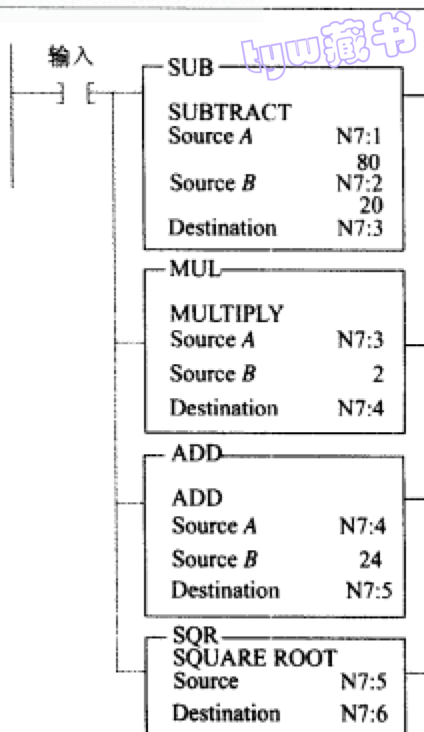

输入
SUB
SUBTRACT
Source A N7:1
80
Source B N7:2
20
Destination N7:3
MUL
MULTIPLY
Source A N7:3
Source B 2
Destination N7:4
ADD
ADD
Source A N7:4
Source B 24
Destination N7:5
SQR
SQUARE ROOT
Source N7:5
Destination N7:6

图 11-31

第 12 章　顺序器指令和移位寄存器指令

学习目标：

- ❑ 确认和描述出各种类型的机械顺序器，并解释其基本工作原理。
- ❑ 解释与 PLC 顺序输入、输出和负载指令相关的信息。
- ❑ 比较事件驱动顺序器和时间驱动顺序器在操作上的不同。
- ❑ 描述位移位寄存器和字移位寄存器的工作原理。
- ❑ 解释并设计用移位寄存器实现的 PLC 程序。

本章包括：（1）PLC 顺序器指令和移位寄存器的操作原理；（2）如何在控制程序中使用这些指令。PLC 的顺序器指令由机械鼓形开关发展而来，但是它处理复杂的顺序控制问题比鼓形开关更容易。移位寄存器通常通过转移数据文件的状态或者数据，跟踪自动化生产线上面的产品部件。

12.1　机械顺序器

顺序器指令是以它所取代的机械序列开关的名字命名的。这些开关通常指的是鼓形开关、旋转开关、步进开关或凸轮开关，如图 12-1 所示，另外还有顺序器开关。图 12-2 所示为一个凸轮开关的操作原理，凸轮通过一个电机驱动，一系列的弹簧片连接到凸轮，因此凸轮在不同的旋转角度，不同的触点会闭合和打开，从而接通和关断不同的电路设备。

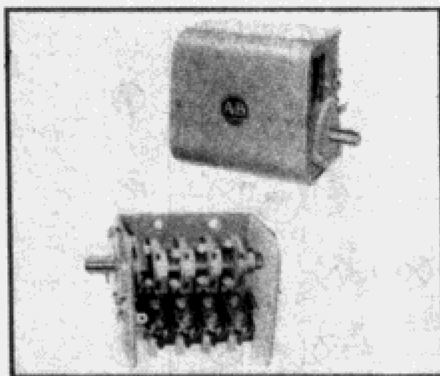


图 12-1　典型的工业旋转凸轮限制开关是控制电路设备，用于当机械设备有重复的周期性操作时（得到 Allen-Bradley 许可）

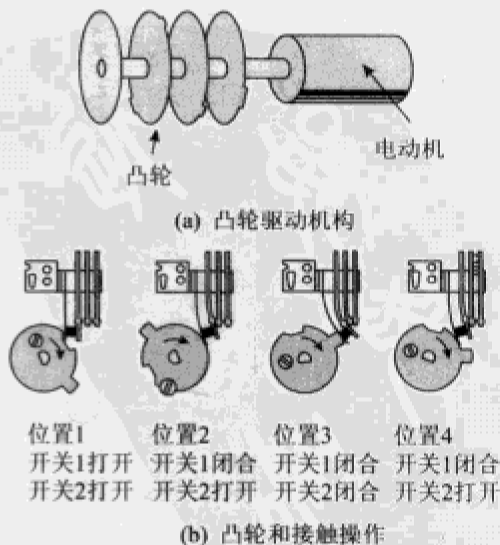


图 12-2　机械凸轮操作顺序器

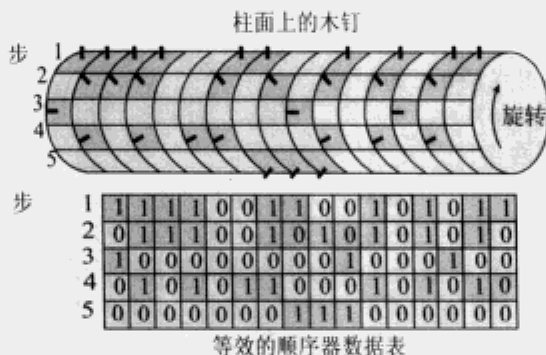
图 12-3 显示的是一个典型的机械凸轮操作顺序器开关。开关包含了一系列触点，这些触点由电机驱动的鼓盘上面的木钉操作，木钉可以随机地放置在鼓盘的周边来操作触

点。当鼓盘旋转时，对准木钉的触点就会闭合，而没有木钉的触点保持打开状态。在这个例子中，出现木钉的地方可以被认为是逻辑 1 或 on，没有木钉的地方是逻辑 0 或 off。

图 12-3 也显示了与鼓形圆柱体等效的顺序器数据表。如果将鼓形圆柱上的前五步取下并展平，就会与下面的表格一样。每个有木钉的水平位置用 1（on）表示，没有木钉的位置用 0（off）表示。

323 ~ 324

顺序器开关用于需要重复操作模式的场合。一个典型的例子就是顺序器开关通过控制每个清洗周期来控制洗碟机（如图 12-4 所示）。周期都是一样的，每一步在特定的时间发生。家用清洗机是使用顺序器的另一个例子，如干燥机和类似的由时钟控制的设备。



步																
1	1	1	1	1	0	0	1	1	0	0	1	0	1	0	1	1
2	0	1	1	1	0	0	1	0	1	0	1	0	1	0	1	0
3	1	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0
4	0	1	0	1	0	1	1	0	0	0	1	0	1	0	1	0
5	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0

图 12-3　机械凸轮操作顺序器

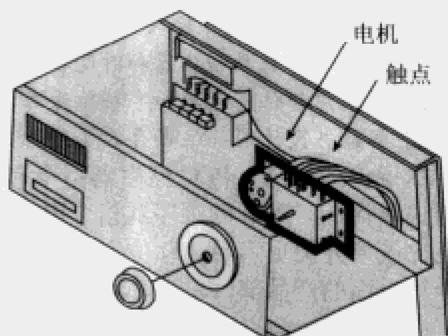
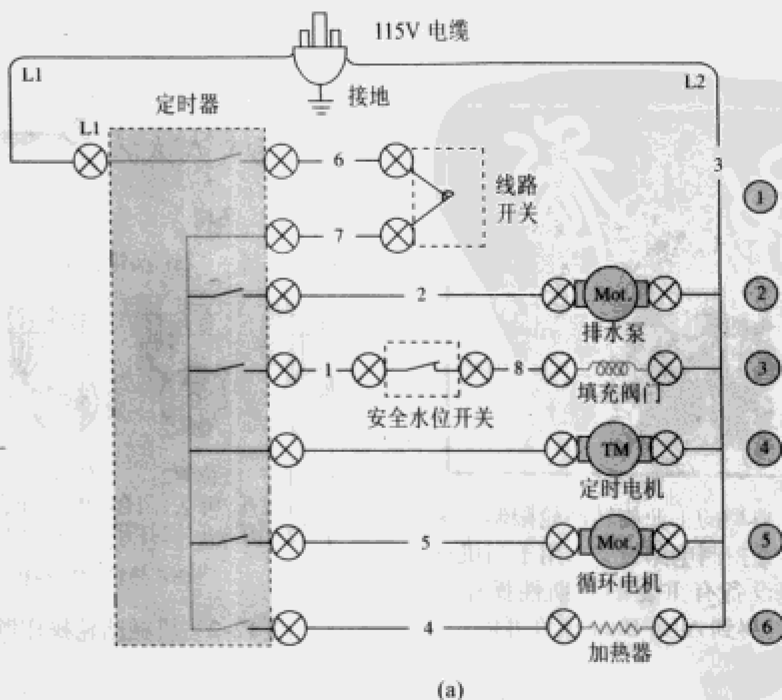


图 12-4　洗碟机顺序器开关

图 12-5 所示为使用凸轮操作顺序器（通常称为定时器）的洗碟机的接线图和数据表。在这个单元中，同步电动机带动机械轴转动，按次序驱动一系列凸轮。凸轮进档的时间增量为 45s。通常定时电机在操作周期中连续运作。图 12-5 的数据表显示了定时器的操作顺序。总共 60 个 45s 的步完成 45 分钟的操作循环。可以在框图中找到"激活电路"一栏中的数字。

325



(a)

图 12-5　洗碟机接线图和数据表

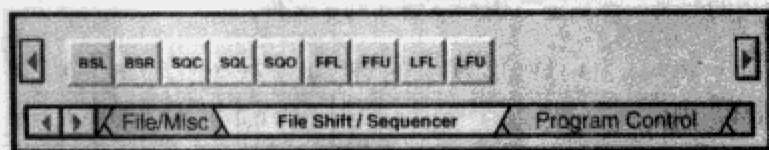
机器功能		定时器增量	激活电路
关断		0-1	
第一次预清洗	排水	2	1 2 4
	填充	3	1 3 4 5
	清洗	4-5	1 4 5 6
	排水	6	1 2 4 5
预冲洗	填充	7	1 3 4 5
	冲洗	8-10	1 4 5 6
	排水	11	1 2 4 5
第二次预清洗	填充	12	1 3 4 5
	清洗	13-15	1 4 5 6
	排水	16	1 2 4
冲洗	填充	17	1 3 4
	冲洗	18-30	1 4 5 6
	排水	31	1 2 4 5
第一次清洗	填充	32	1 3 4 5
	清洗	33-34	1 4 5 6
	排水	35	1 2 4 5
第二次清洗	填充	36	1 3 4 5
	清洗	37-41	1 4 5 6
	排水	42	1 2 4 5
烘干	烘干	43-58	1 4 6
	排水	59	1 2 4 6
	烘干	60	1 4 6

(b)

图 12-5　（续）

12.2　顺序器指令

PLC 的顺序器指令可以代替电子机械鼓形顺序器或鼓形开关。顺序器指令和鼓形开关一样能连续的重复实现输出特定的通/断模式，但是 PLC 的顺序器指令更加灵活。图 12-6 所示为基于 Allen-Bradley SLC-500PLC 和它相关软件的典型的顺序器指令。顺序器指令可以允许使用单个指令或两个指令来实现非常复杂的操作，从而简化了梯形图程序。



命　令	名　称	说　明
SQO	顺序器输出指令	通过一个屏蔽字，传送 16 位数据到映像地址，控制输出，实现控制机器的顺序操作
SQC	顺序器比较指令	通过一个屏蔽字，传送 16 位数据到映像地址，监视输入，实现控制机器的顺序操作
SQL	顺序器装入指令	按照操作次序手动顺序操作机器获取参考条件

图 12-6　基于 Allen-Bradley SLC-500PLC 和它相关软件 RSLogix 的顺序器指令

顺序器指令是一种典型的用于控制自动化装配机器的指令，其操作具有一致性和重复性。顺序器指令可以大大简化很多应用程序。例如用顺序器指令对 16 个离散输出进行通/断控制，只需要一个梯级就能满足要求，相比之下功能相当的触点线圈梯形控制结构却需要 16 个梯级。

326

在当今的 PLC 中，顺序器指令是一个功能强大的指令。一个可编程顺序器可以取代机械鼓形开关。对顺序器进行编程时，要输入二进制信息到一系列存储字中。这些存储字被称为字文件。顺序器的每一步都要先输入数据到字文件，当顺序器顺序执行每一步时，二进制信息被从字文件传送到输出字中。

顺序器输出（SQO）指令用于顺序控制输出设备。期望的操作次序存储于一个数据文件中，然后这个信息被顺序传送到输出。图 12-7 显示的是顺序器输出指令是如何工作的，在这个例子中，16 个指示灯作为输出。每个灯代表输出字 050 的一个位地址（从 1 到 16）。这些灯在一个 4 步的序列中，模拟两路交通灯。

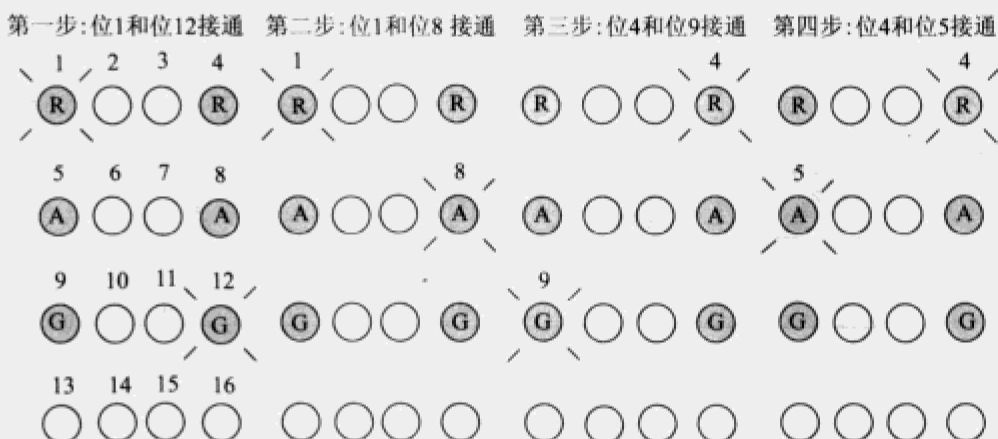
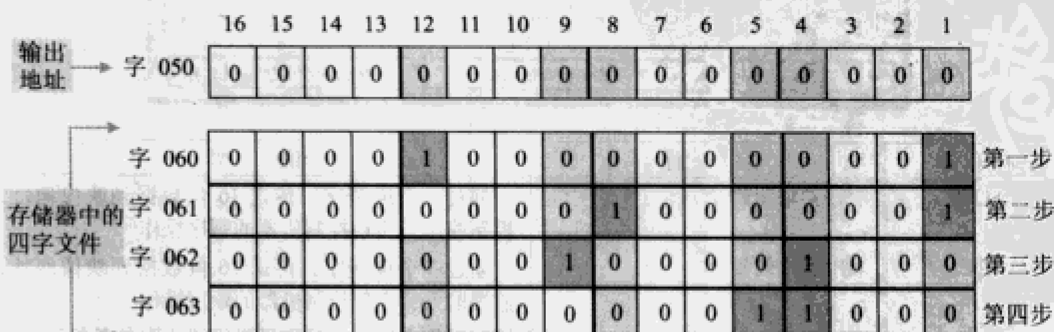


图 12-7 顺序器的步（R－红灯，G－绿灯，A－黄灯）

每一顺序步的数据被输入到字文件，如图 12-8 所示。在这个例子中，字 60、61、62 和 63 作为四字文件。通过编程器，反映期望亮灯顺序的二进制信息（1 或 0）被输入到文件中的每一个字。为方便编程，有些 PLC 允许输入到字文件的数据为八进制，十六进制，BCD 码或类似的数制。如果是这种情况，每个顺序步所需的二进制信息，必须先要转换为 PLC 能够使用的数制，然后通过编程器输入到字文件。



		16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	
输出地址	字 050	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
存储器中的四字文件	字 060	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	第一步
	字 061	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	第二步
	字 062	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	第三步
	字 063	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	第四步

图 12-8 每个顺序器步的二进制信息

一旦将信息输入到顺序器的字文件，PLC 就准备对输出灯进行控制。当顺序器被接通并进入第一步，文件字 060 中的二进制信息被传送到输出字 050。这样灯 1 和灯 12 被接通，其余的关断。进入第二步时，把字 061 中的数据传送到输出字 050。这样灯 1 和灯 8 接通，其余的关断。进入第三步时，把字 062 中的数据传送到输出字 050，灯 4 和灯 9 被接通，其余的关断。当进入第四步时，把字 063 中的数据传送到字 050，灯 4 和灯 5 被接通，其余的灯关断。当执行完最后一步，顺序器通过自动或手动的方式复位到第一步。 327

当顺序器对整个输出字进行操作时，可能字中的有些字的输出位不需要顺序器来控制。在这个例子中输出字中的位 2、3、6、7、10、11、13、14、15 和 16 没有被顺序器使用，但可能在程序中的其他地方使用。

为了防止顺序器对输出字的这些位进行控制，采用了一个屏蔽字（040），如图 12-9 所示。屏蔽字有选择性地屏蔽了由顺序器字文件传送到输出字的数据。对于顺序器要进行控制的输出字 050 中的每一位，与之对应的屏蔽字 040 中的位必须设置为 1。输出字 050 的其他位设置为 0，这样就能独立于顺序器使用了。

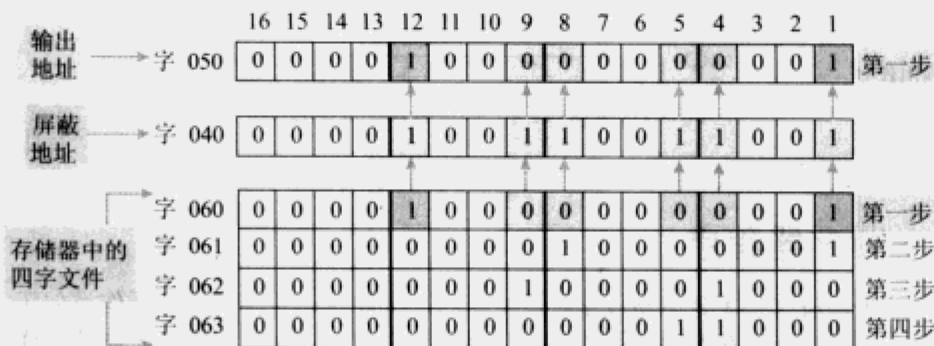


图 12-9 使用一个屏蔽字

顺序器和其他 PLC 指令一样，编程因 PLC 不同而有区别，但是原理都是一样的。顺序器编程与传统的编程相比，优势在于大大节省了存储字。一般传统的标准编程方式需要 100 个字，而顺序器指令只需 20 或更少个字就能满足功能要求。而且顺序器通过设置事件的顺序简化了编程，也更便于以后更改。

顺序器指令通常都是保持型的，一条单独的指令可以执行的外部输出和顺序步的数量都有限制。许多顺序器指令在完成最后一步时，都自动复位到第一步。另外一些指令提供单独的复位控制线或二者的联合。 328

图 12-10 所示为一个顺序器输出（SQO）指令的典型样式。一条单独的指令确定了输出模式数据的存储位置（文件#B3:0），输出数据的目的地址（字 O:2）和顺序器步的长度或数量（4）。这个指令也管理和跟踪当前顺序器的位置。每当指令前面的条件状态逻辑由假变为真时，顺序器就会增加到下一步。当顺序器文件的传送完最后一个字，完成位置位，下一次上升沿到来时指令复位到第一步。启动时，当处理器从程序模式变为运行模式时，顺序器的操作取决于初始扫描周期梯级为真或假。当有效的真输入启动了顺序器指令，顺序器就会在 0 位置，当输入为假时顺序器就在 1 位置。 329 330

屏蔽字本章前面已经作了解释，它是顺序器文件中的数据在被传送到目的地址或输出字之前，必须通过的过滤器。只有当屏蔽字中的位设置为 1 时才允许通过数据到目的地址。要输入十六进制的屏蔽字，在数据后要加字母 h（十六进制）；要输入二进制屏蔽字，后面要加字母 b（二进制）；要输入十进制屏蔽字，直接输入即可。

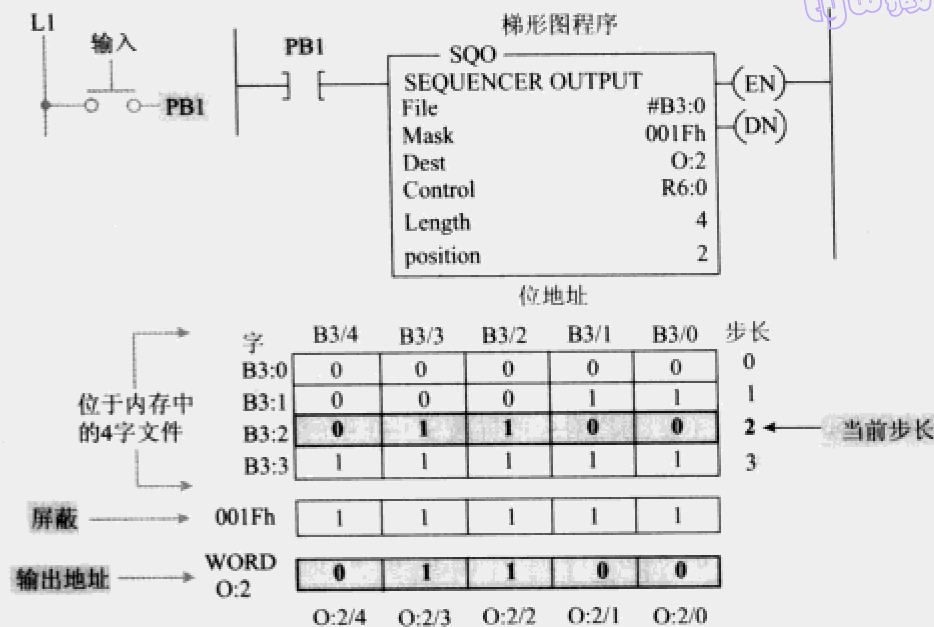


图 12-10　典型的顺序器输出指令

12.3　顺序器程序

顺序器程序可以是事件驱动或是时间驱动的。事件驱动的顺序器的操作类似于一个机械步进开关，每当有加在上面的脉冲时，就会增加一步。时间驱动的顺序器的操作类似于机械鼓形开关，经过一段预置的时间后自动增加一步。

顺序器图表如图 12-11 所示，其中显示由顺序器指令控制的输出的操作次序。表格采用矩阵形式，矩阵就是二维、成矩形的数组。时间驱动的顺序器图表的横轴通常表示输出，纵轴表示时间。事件驱动的顺序器横轴表示输出，纵轴表示输入或事件。

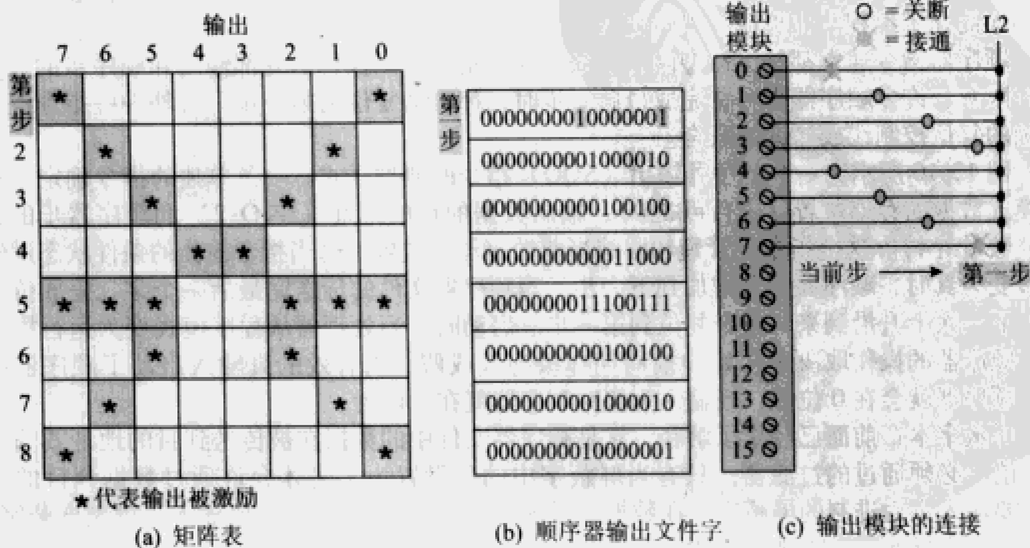


图 12-11　顺序器图表

图 12-12 所示为用于控制十字路口交通灯的时间驱动顺序器程序。在这个例子中，交通灯的控制用两个顺序器输出（SQO）指令和一个单独的通延时定时器（TON）指令实现。第一个顺序器文件（#N7:0）设置交通灯变化的四种状态。文件中的信息由程序被传送到输出（O:2）。第二个顺序器文件（#N7:10）包含了预设时间值（25s 和 5s），程序中把这个文件中的信息传送到定时器 T4:1 的预置值中。屏蔽字允许需要的数据通过，而阻止不需要的数据。

331

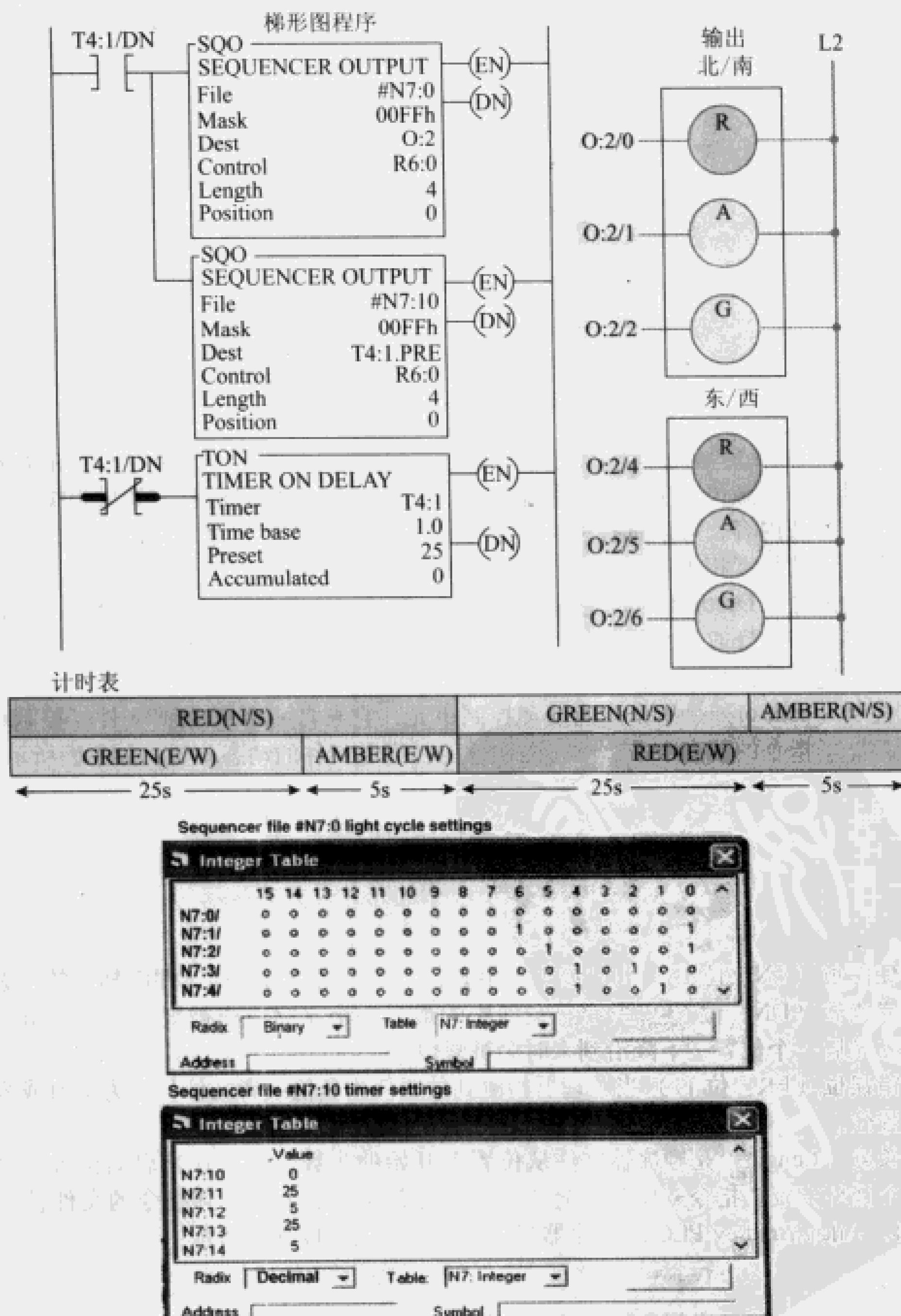


图 12-12　用于实现十字路口交通灯程序的时间驱动顺序器指令

虽然顺序器指令的编程根据 PLC 模块和生产厂家不同而有区别，但是编程的原理是一样的。顺序器控制的事件的次序，由每个连续字的模式和字在队列中的序号决定。顺序器输出（SQO）指令是 Allen-Bradley PLC-5 和 SLC-500 指令系统的一部分，如图 12-13 所示。下面的参数和地址必须输入到指令块中：

332

- **文件**（File）是顺序器文件的地址，必须用文件标志符#来标识。当指令发生上升沿跳变时，文件中包含的数据会传送到目的地址。文件中的每个字代表一个位置，从位置 0 开始，一直到文件长度的位置。实际的文件长度比指令中显示的文件长度多一个字。

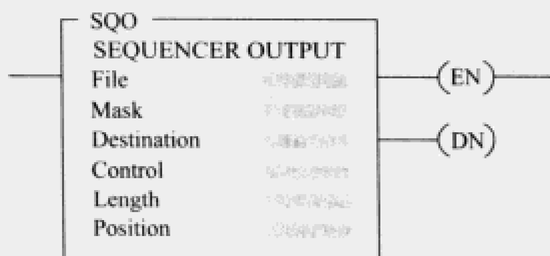


图 12-13 Allen-Bradley PLC-5 和 SLC-500 顺序器输出（SQO）指令

- **屏蔽**（Mask）是一个十六进制码或是屏蔽字/文件的地址，指令传送的数据要先通过它。屏蔽位为 1 时允许数据通过，为 0 时阻止数据通过。如果根据应用需要改变屏蔽内容，可以选择使用屏蔽字或文件。如果屏蔽为文件，它的长度和顺序器文件的长度相等，两个文件自动匹配。
- **目的地址**（Destination）是输出地址，数据从文件复制到该输出地址中。目的地址可以是字地址或是文件地址。如果是一个文件地址，文件地址的位置和目的地址的位置自动地相同。大多数的应用中，目的地址为一个字地址。
- **控制**（control）是指令的地址和控制单元，它来自一个 R 数据文件。控制单元中存储了指令的状态字节、顺序器文件的长度和文件的动态位置，如下面所示：

	15	13	11	08	00
字 0	EN	DN	ER	FD	
字 1	顺序器文件的长度				
字 2	位置				

333

- **使能位**（EN，位 15）由梯级上升沿置位，指示指令被使能，跟随梯级的状态。
- **完成位**（DN，位 13）当指令文件中的最后一个字完成时被置位。如果指令工作在最后一个位置，下降沿到来时会被复位。
- **错误位**（ER，位 11）当处理器检测到位置值为负值或者文件长度为负值或零时被置位。
- **长度**（Length）是顺序器文件从位置 1 开始的步数，位置 0 是启动位置。每当一个循环完成，指令复位到位置 1。实际的文件长度为输入到指令的文件长度值加 1。Allen-Bradley PLC 的顺序器指令文件的最大长度如下所示：

PLC-5	1-1000
SLC-500	1-255
MicroLogic 1000	1-104

- **位置**（position）是顺序器文件中的字存储区或步，指令从其中传送数据。可以输入任意的小于文件长度值的值，但是当指令完成最后一个位置，上升沿到来时，位置值复位为1。在启动顺序之前，我们需要一个顺序器处于中间位置的启动点。用全0表示这个中间位置，因此在位置0所有输出关断。

图12-14的SLC-500的事件驱动顺序器输出程序，可以用来解释顺序器输出指令的操作过程。当输入*A*的上升沿到来时，屏蔽字中位为1所对应的文件#3:0中的数据被传送到目的地址O:2中。位置单元中设置了一个位置，文件中相应位置字数据被复制。当位置达到最后一个位置，指令的下一次下降沿到来，位置被复位为1。位置0在下列情况下被执行：位置为0、指令为真并且处理器处在运行模式。位置0通常作为一个启动位置，通过程序设置位置为0。当指令遇到上升沿时，位置变为1并复制文件中第一个位置的数据到目的地址。在后面的次序中，又会重新复位到位置1。注意到O:2中的数据，除了位10～位13以外，与文件中的第二个位置中的数据相同。10～13这些位可以由程序中其他地方控制，因为屏蔽字中这些位相应的值为0，所以顺序器指令不起作用。由于顺序器指令执行每一步时都要自动地增加，因此在程序中要结合一个定时器。图12-15所示的定时器预置值为3s，作为顺序器的输入脉冲。每一步发生后，顺序器的使能位复位定时器。当输入*A*闭合时，电路自动地完成顺序器的8步操作，时间间隔为3s。

334

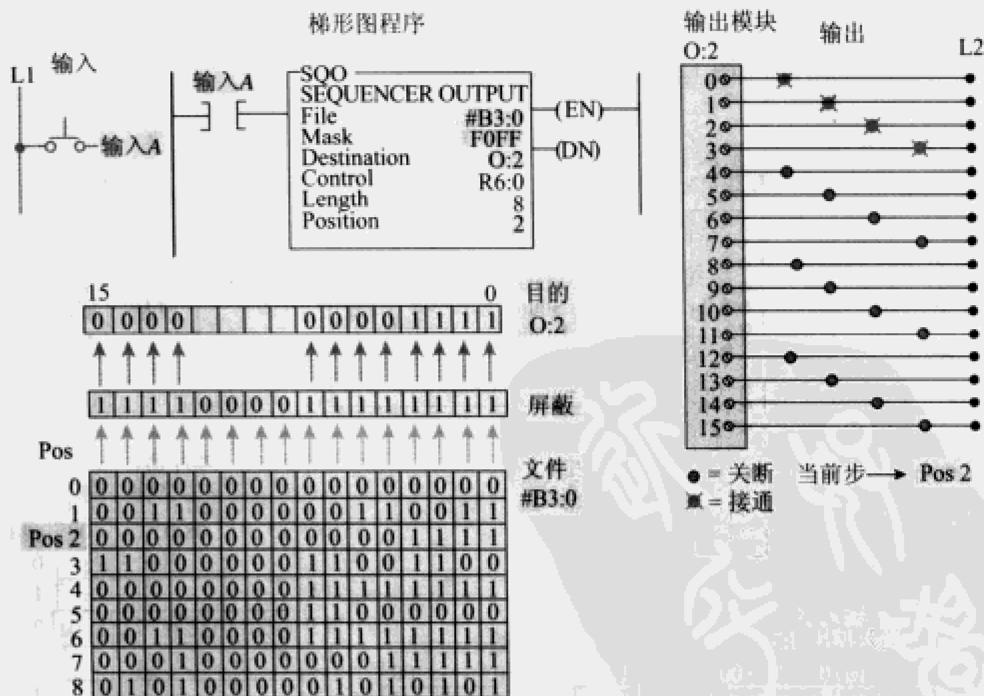


图12-14 SLC-500的事件驱动顺序器输出指令程序

顺序器输入（SQI）指令在PLC中应用也很普遍。这个指令用于比较输入数据与存储在顺序器文件中的数据。例如它可以比较输入设备的状态和期望状态，如果状态相同，则指令为真。

图12-16所示的PLC-5顺序器输入程序，可以用来解释顺序器输入指令的操作原理。指令的输入参数，除了目的地址用源地址代替以外，其他的与顺序器输出指令类似。SQI指令把文件I:3的输入映像数据，通过屏蔽字FFF0，与#N7:1中的参考数据相比较，看其

是否相等。当这些字中没有被屏蔽掉的所有数据位与参考数据中对应的位相等时，指令为真；否则指令为假。输入数据可以指示输入设备的状态，例如本例中的输入开关的状态组合。当打开和关闭的组合状态与顺序器参考文件的一个步中的 1 和 0 的组合相同时，接通顺序器指令的输出 PL1。

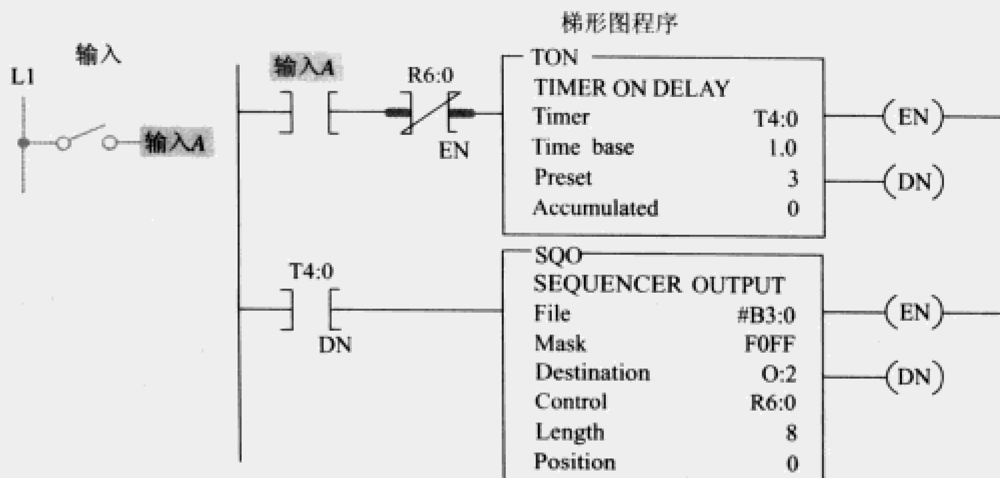


图 12-15 事件驱动顺序器输出指令程序

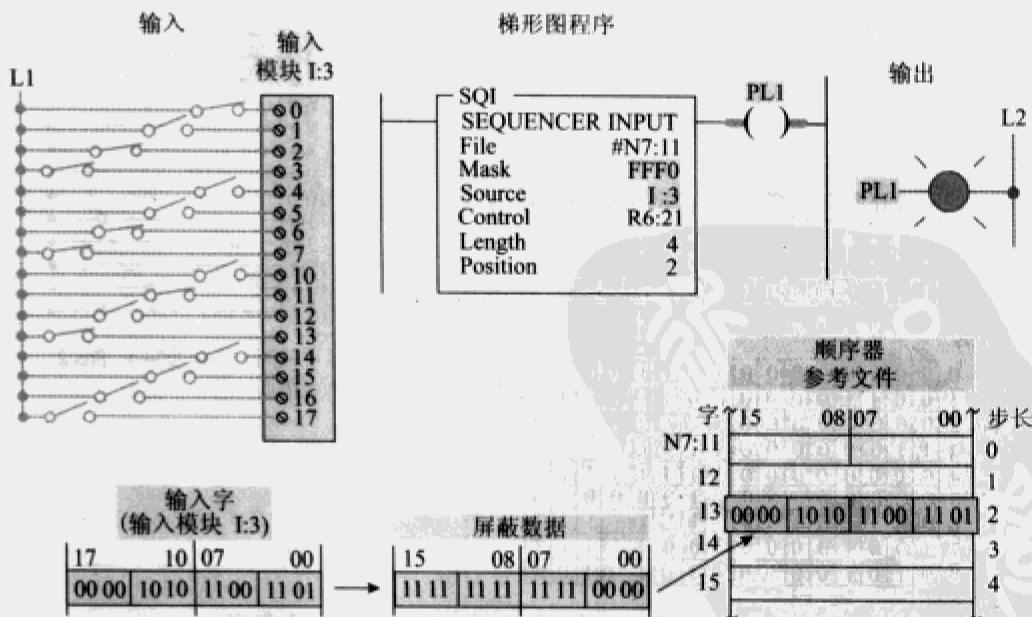


图 12-16 顺序器输入（SQI）指令程序

335

和顺序器输出指令不同的是，当它的控制逻辑上升沿到来时，顺序器输入指令（SQI）不能自动增加位置。SQI 控制单元中的位置值必须通过另一个指令来改变（如传送指令），然后选择新的数据文件值与源地址中的数据比较。

图 12-17 所示的程序使用顺序器输入指令（SQI）和顺序器输出指令（SQO）组合成一对，分别监视和控制一个顺序操作。对这对的顺序器输入和输出指令进行编程时，使用

相同的控制地址、长度和位置。由于控制单元中的地址相同（R6:5），本例中通过顺序器输出指令来指示顺序器输入指令。这种类型的编程技术使顺序器输入和顺序器输出指令同步工作，当有特定次序的输入发生时，引起特定次序的输出。

梯形图程序

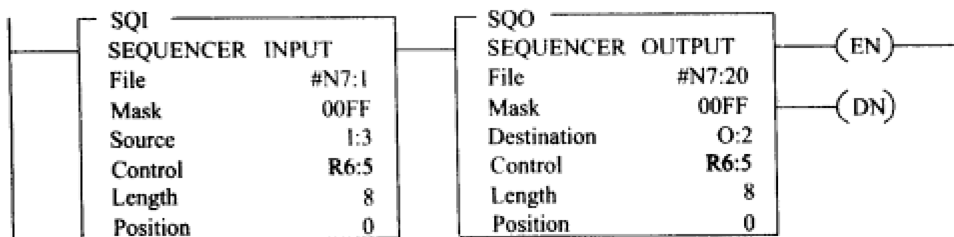


图 12-17 PLC-5 顺序器输入和顺序器输出指令组合

Allen-Bradley PLC-5 的顺序器输入（SQI）和 SLC-500 的顺序器比较（SQC）指令相似，但不完全相同。两个指令都是通过屏蔽比较源地址中的数据和顺序器文件字中的数据，并且都是当它们的控制逻辑为真时，才执行比较功能。SQI 指令和 SQC 指令输入文件的第一个数据字是作为启动状态的，不包含在后面的重复循环比较次序中。 336

这两个指令的主要区别在于，SQC 指令在 SLC-500 程序中被看作是输出单元，而 SQI 指令在 PLC-5 程序中被看作是输入单元。不同于 SQI 指令，当控制逻辑由假变真时，SQC 自动增加位置。SQC 作为输出单元不能再去控制另一个输出单元，因此在控制单元中增加了一个附加发现位（FD）。当源地址的输入状态与参考顺序器文件中的状态匹配时，FD 位被置位；如果不匹配则 FD 位为假。控制 SQO 和 SQC 指令的控制逻辑可以检测 FD，因此它们是一起增加的。例如图 12-17 所示的 PLC-5 程序，用 SLC-500 指令重新编写后如图 12-18 所示。

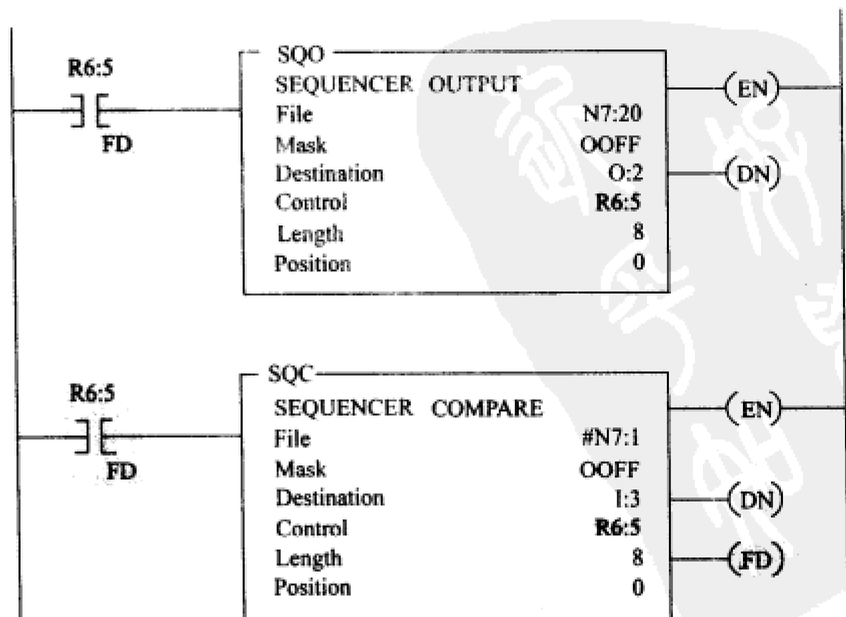
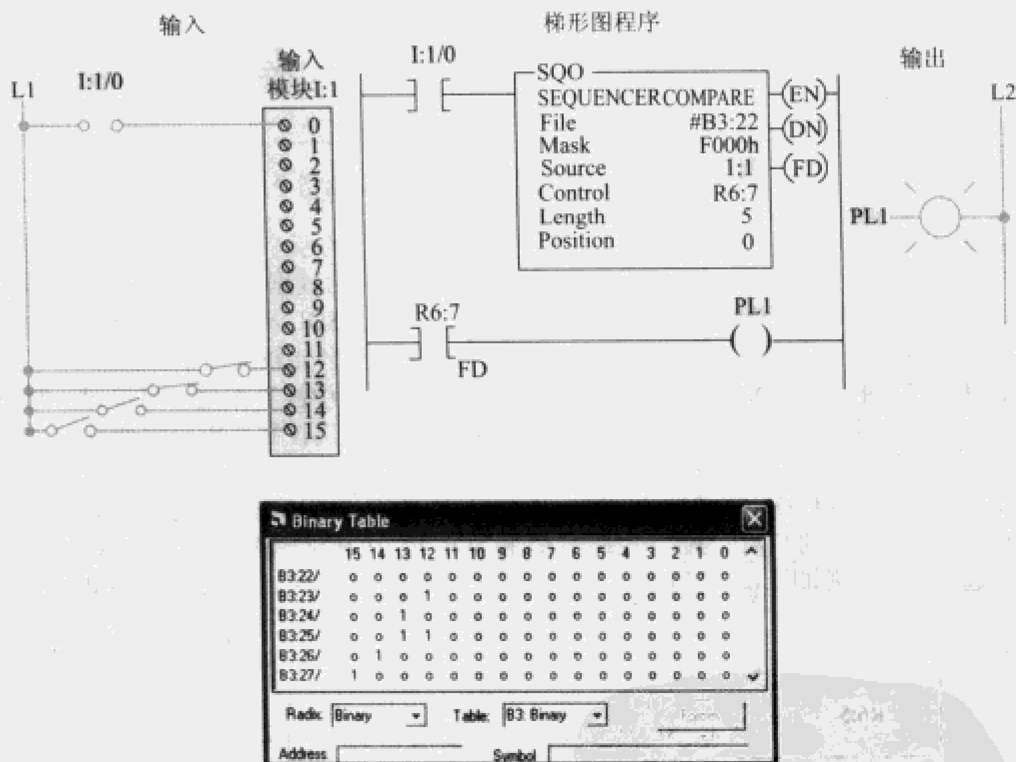


图 12-18 SLC-500 顺序器比较和顺序器输出指令组合

图 12-19 为 SLC-500 顺序器比较指令（SQC）程序的一个例子。源地址 I:1 中的高 4 位数据与文件 B3:22 中的数据比较。在本例中 I:1 中的高 4 位数据与#B3:22 中的高 4 位在位置 3 时匹配。如果这时输入按钮 I:1/0 为真，发现位（FD）被置位，接通 PL1。只要连接到 I:1/12、I:1/13、I:1/14 和 I:1/15 开关的接通和关断状态组合，在某一步中与顺序器参考文件中的 1 和 0 状态匹配，并且输入 I:1/0 为真，就会接通输出 PL1。注意屏蔽（F000h）是怎样实现把顺序器没有用到的位单独使用的。在这个例子中，没有用到的位 I:1/0 用于顺序器比较梯级的状态输入。



	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
B3:22/	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
B3:23/	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
B3:24/	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
B3:25/	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0
B3:26/	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
B3:27/	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

图 12-19　SLC-500 的顺序器比较指令（SQC）程序

Allen-Bradley PLC 还有一个顺序器装入（SQL）指令，功能类似于从字传送数据到文件中。它可以作为一个工具一次一步的将数据输送到顺序器文件中。例如机器的装配可以按照它的操作次序手动进行操作，输入设备按步读取数值。每步输入设备的状态被写入到顺序器比较指令的数据文件中。因此文件中装入了每一步所期望的数据状态，然后这些数据就可以在自动运行状态下，与输入设备状态进行比较。

337

图 12-20 所示的 SLC-500 顺序器装入程序，可以用来解释顺序器指令的操作原理。顺序器装入指令用于将数据装载到文件，它在机器正常运行时不起作用。它代替了用编程设备手动转载数据到文件。顺序器装入指令没有设置屏蔽，它将数据从源地址复制到文件。当指令状态从假变为真时，指令步入下一个位置并复制数据。当指令完成最后一个位置时，产生一个下降沿，指令复位到第一步。它在位置 0 传送数据的条件为：处于位置 0、指令为真并且处理器处于运行模式。手动装配机器时，连接到源地址 I:2 的开关状态可以在每个位置被读取，并且在按下 PB1 的瞬时被写入文件。否则数据只能手动输入到文件。

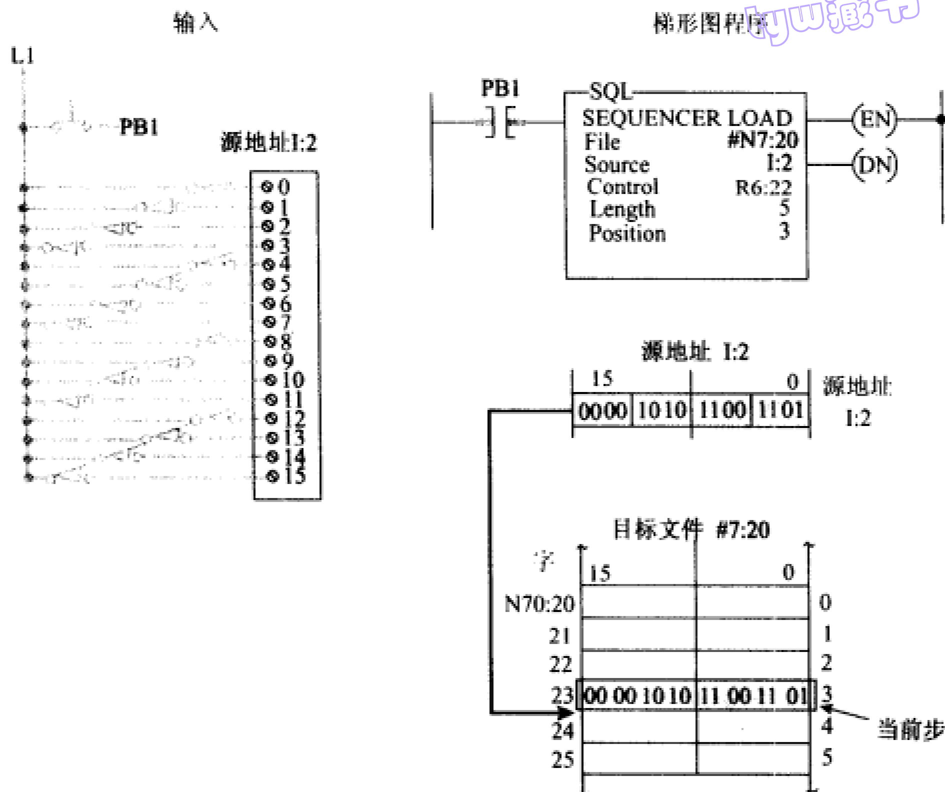


图 12-20 SLC-500 顺序器装入（SQL）指令程序

12.4 移位寄存器

PLC 不仅使用固定形式的寄存器（字）位，还可以方便地处理并改变特殊位。位移位寄存器允许数据位在一个单独的寄存器或一组寄存器中转移。位移位寄存器中位的转移是按次序串行移动的（从位到位）。

移位寄存器可以用来模拟或跟踪部件和信息的移动。当我们需要存储一个事件的状态，以备后继处理的时候，就可以采用移位寄存器。移位寄存器可以通过数据文件转移状态或数据值。通常移位寄存器的应用包括以下几方面：[338]

- 跟踪装配生产线上面的部件
- 控制机器或过程操作
- 存储控制
- 系统诊断

图 12-21 所示为移位寄存器的基本原理。通常由移位脉冲或者时钟信号，引起移位寄存器中每位向右移动一位。在有些情况下，输入到移位寄存器的数字会超过寄存器的存储容量。如果出现这种情况，装入移位寄存器的第一个数据位，在移位寄存器的最后面丢失。通常，移位寄存器中的数据可以代表如下信息：[339]

- 部件类型、质量和大小
- 部件的有无
- 事件发生的顺序
- 数字或位置的识别
- 导致停机的事故状态信息

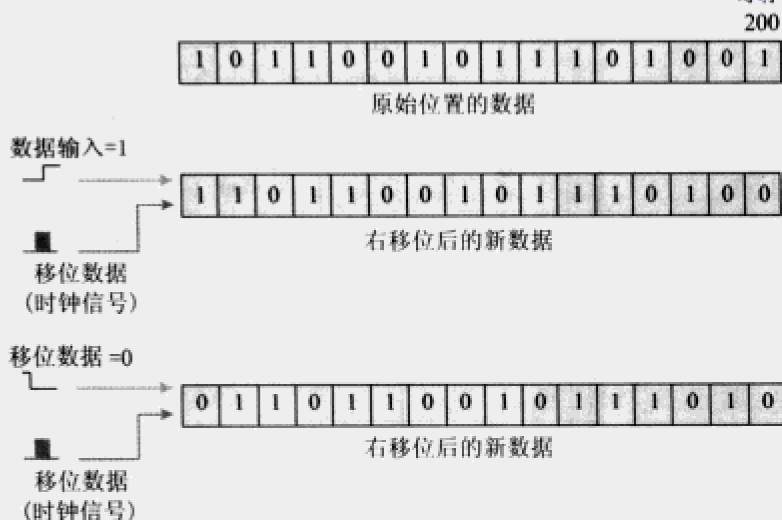


图 12-21　移位寄存器的基本原理

可以通过移位数据文件中的状态数据，实现移位寄存器的右移或左移状态的编程。如果要跟踪状态基础上面的元件，可以采用位移位寄存器。位移位指令，通过数据文件，移位一个源位地址中的位状态，并将其送到卸载位，每次移动一位。有两种位移位指令：位左移（BSL）指令（在数据文件中从低地址到高地址移位）和位右移（BSR）指令（在数据文件中从高地址向低地址移位），如图 12-22 所示。有些 PLC 提供循环移位寄存器功能，可以连续不断地重复操作。

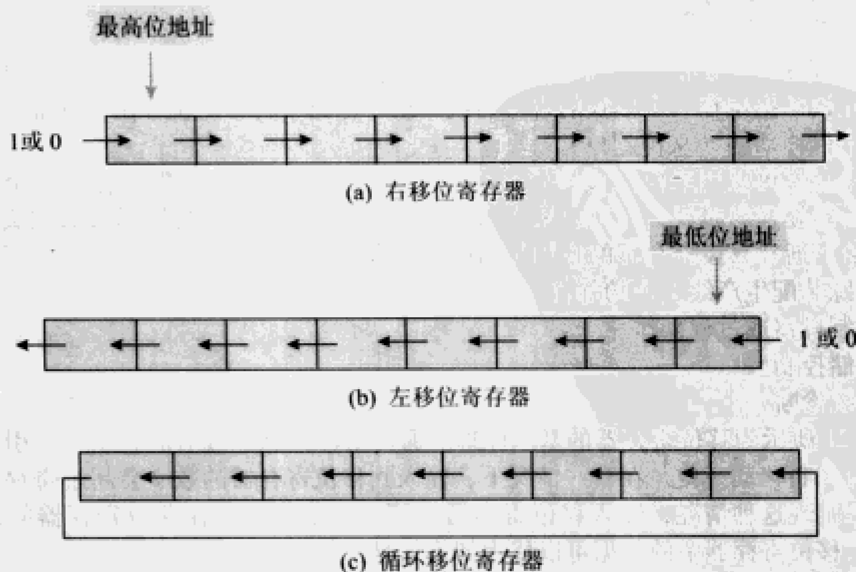
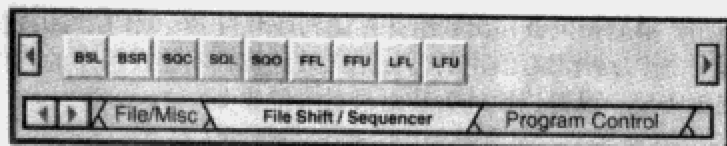


图 12-22　位左移和位右移寄存器指令

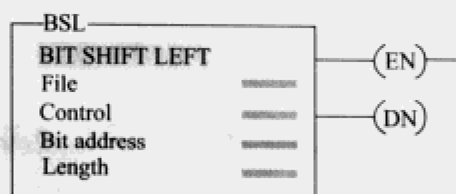
当使用位移位寄存器指令时，可以通过在寄存器中的位置识别每一个位。因此对每个位的操作就变为识别它在寄存器中的位置，而不是传统的字或位号地址形式。

图 12-23 所示为 Allen-Bradley PLC-5 和 SLC-500 控制器指令集中的位左移（BSL）和位右移（BSR）指令。用于移位寄存器的数据文件通常为位文件，这是因为它的数据以二进制形式显示，以便于阅读。BSL 和 BSR 指令为输出指令，一次只装一位数据到位数组。数据在数组中移位，然后一次卸载一位。

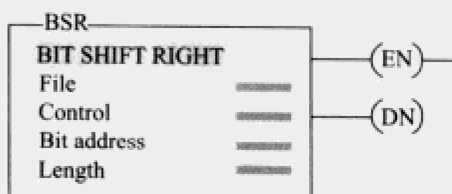


命　　令	名　　称	说　　明
BSL	位左移指令	将一位装入数据数组，在数组中左移数据，数组中最后一位被卸载
BSR	位右移指令	将一位装入数据数组，在数组中右移数据，数组中最后一位被卸载

(a) 基于Allen-Bradley PLC-5和SLC-500PLC及其编辑软件RSLogix的位左移和位右移指令



(b) 位左移指令



(c) 位右移指令

图 12-23　Allen-Bradley PLC-5 和 SLC-500 的位左移和位右移指令

移位寄存器指令使数据位在 PLC 的寄存器内部或寄存器之间移动。数组就是存储区中多个数据字（大于 16 位）的集合。由于 BSL 和 BSR 指令一次只移动一位，因此被移位的数据存储在二进制或位的文件中。当多个字组合到一起，这个集合就被称为数组。用于移位寄存器的数据文件或数组，一般包含的数据都超过 16 位。

340

对一个位移位指令编程，需要给处理器提供以下信息：

- **文件**（File）是要处理的位数组的地址。数组必须从一个 16 位开始，例如采用字 1、2 和 3 等的位 0。可以在小于文件最大长度内的任何位号处来结束数组。超过数组中最后一位（由长度决定）和达到下一个字的位都被视为无效。
- **控制**（Control）是控制结构的地址，控制指令的操作。该地址不能用于控制其他的指令。组成控制单元的三个字列在下面：

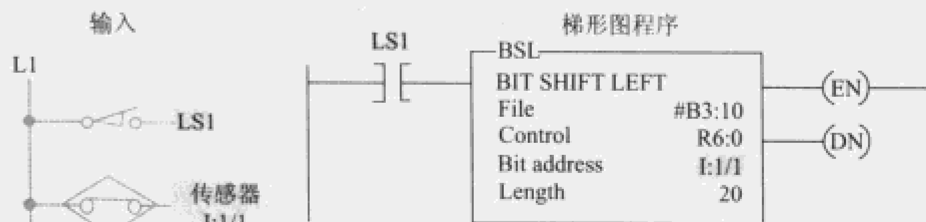
位	15	13	11	10
控制字	EN	DN	ER	UL
长度字	存储文件的长度			
位置字	指示 0 到文件长度之间的当前位			

位 15 为使能位，当指令为真时被置位。位 13 为完成位，当移动完文件 1 位置的所有数据位时，指令被置位。当指令变为假时完成位被复位。位 11 为错误位，当检测到错误时，指令被置位，如当输入的长度为负数时。位 10 为卸载位，是当指令从假变为真时文件中的最后一位的载入位置。当下一次移位发生时，除非用附加的编程来保持这些数据，否则这些数据会丢失。

341

- **位地址**（Bit address）是源数据位的地址。指令插入这个位的位置，对于 BSL 指令是第一位（最低位），对于 BSR 指令是最后一位（最高位）。
- **长度**（Length）是位数组的长度或以位为单位的文件长度。

图 12-24 的位左移程序说明了 BSL 指令的操作原理。限位开关 LS1 的上升沿产生移位脉冲。当梯级从假变为真时，使能位被置位，数据块向左移动。传感位 I:1/1 被移动到第一个位置 B3:10/0。最后一位被移出数组并存储到卸载位 R6:0/UL，卸载位原来的数据丢失。注意文件最后一个字中没有使用的部分是无效地址，不能在程序的其他部分使用。对于循环操作，根据需要，设置位地址为数组最后一位或 UL 位。



(a) 程序



(b) LS1产生移位脉冲前的数组数据



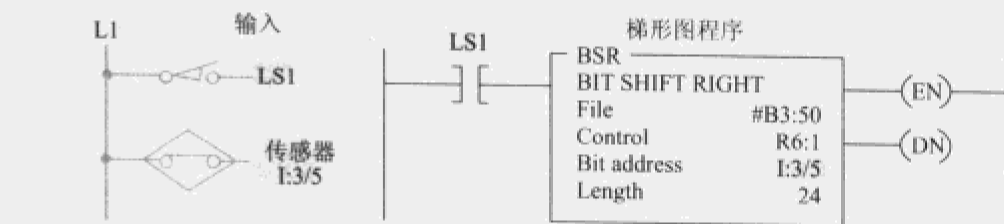
(c) LS1产生移位脉冲后的数组数据

图 12-24　位左移程序

图 12-25 所示的位右移程序可以用来说明 BSR 指令的操作原理。在梯级的上升沿到来之前，字 B3:50 和字 B3:51 的状态如图所示。位 I:3/5 的状态为 0，卸载位 R6:1/UL 的状态为 1。当限位开关闭合，位 I:3/5 的状态转移到 B3:51/7，也就是文件中的第 24 位。文件中的所有位的状态都向右移动了一位。B3:50/0 移动到卸载位 R6:1/UL。卸载位先前的信息丢失。

图 12-26 为由移位寄存器控制的喷漆操作。文件的每一位代表了生产线上的一个位置，位的状态指示了是否有部件。位 I:1/2 检测生产线上是否有部件到来。移位寄存器的功能是跟踪要被喷漆的部件。位左移指令用于指示生产线的正向运动。当部件在传送带上面通过，那些代表了部件在传送带上的移位寄存器位对应的部件被喷漆。LS1 用于检测挂

钩，LS2 用于检测部件。这个操作的逻辑为，当被喷漆的部件和部件挂钩同时出现（通过 LS1 和 LS2 的一起动作来监视）时，逻辑 1 被送入移位寄存器。逻辑 1 引起底漆喷枪的动作，五步后，当移位寄存器又出现逻辑 1，面漆喷枪被启动操作。限位开关 3 对烘箱中的部件计数。在喷涂的最后，限位开关 2 和限位开关 3 获得的计数值相等（PL1 接通），说明进入喷涂线的部件和已经完成的部件数相等。移位寄存器中的逻辑 0 表示传送带上面没有部件需要喷涂，因此阻止了喷枪的操作。



(a) 程序



Binary Table

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
B3:50/	1	0	1	1	0	0	0	1	1	0	0	1	0	1	1	0
B3:51/	0	0	0	0	0	0	0	0	1	0	1	1	0	0	1	1

Radix: Binary Table: B3: Binary Forces

Address Symbol

(b) LS1产生移位脉冲前的数组数据

0 卸载位 R6:1/UL

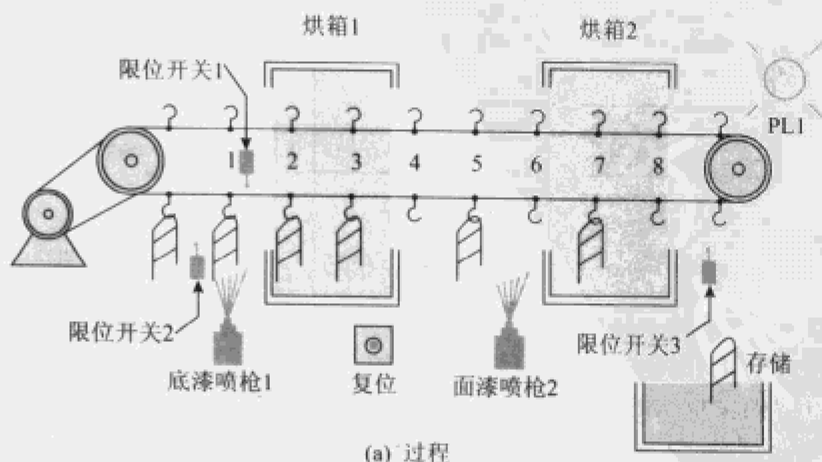
移位方向

1	1	0	1	1	0	0	0	1	1	0	0	1	0	1	1	B3:50
无效								0	1	0	1	1	0	0	1	B3:51

0 24 I:3/5 位地址

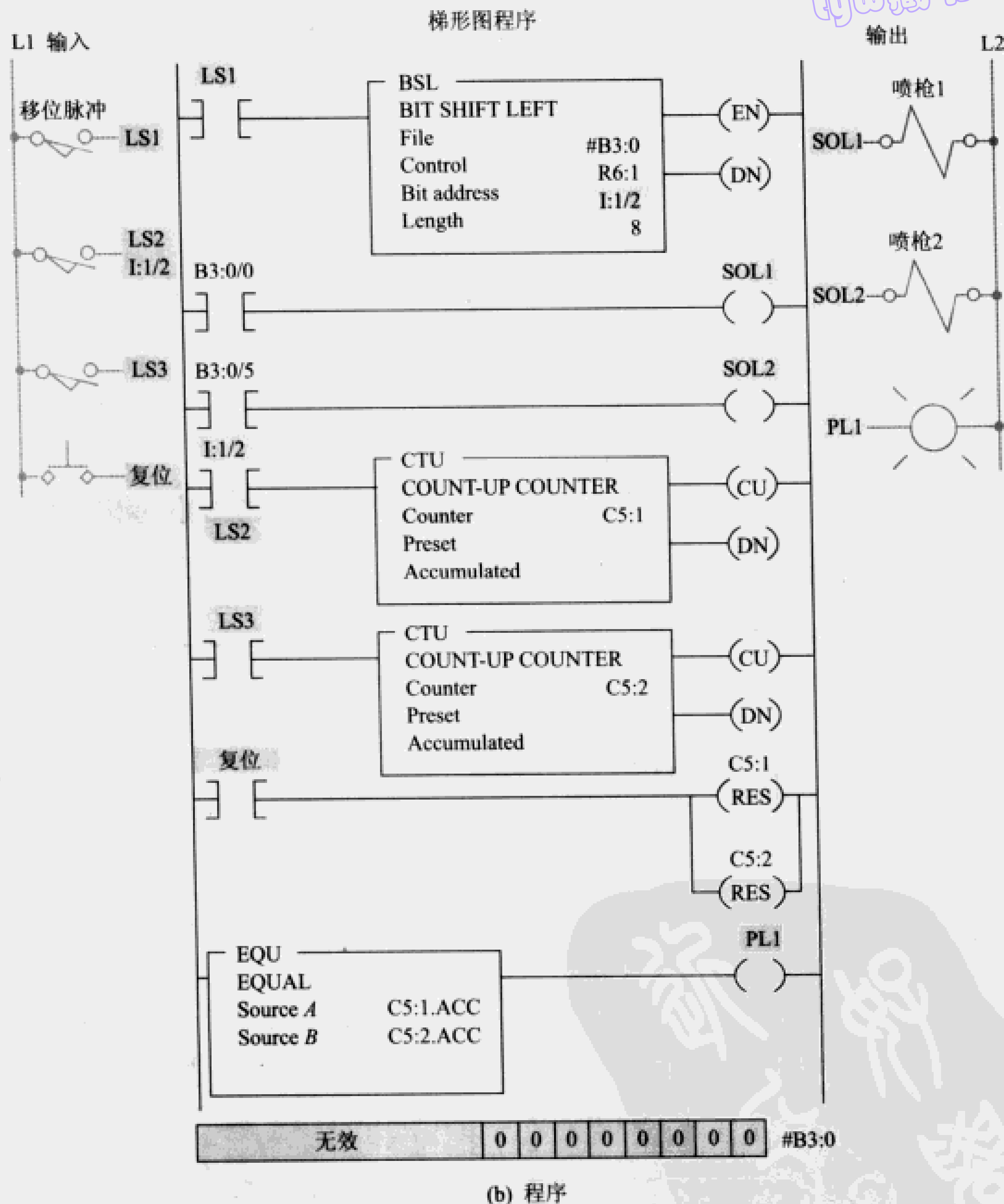
(c) LS1产生移位脉冲后的数组数据

图 12-25 位右移程序



(a) 过程

图 12-26 用于喷漆操作的移位寄存器



(b) 程序

图 12-26 （续）

图 12-27 所示的程序为用一个位移位指令跟踪通过 16 站机器的载体。接近开关 1 用于检测一个载体，接近开关 2 用于检测载体上面的一个部件。当带有部件的载体通过机器时，连接到输出模块 O:4 的指示灯变亮。当空载体通过机器时，指示灯熄灭。第 4 站为检修站。如果部件损坏，产品检查员按下 PB1，把损坏部件从系统中去除，输出指示灯 O:4/4熄灭。在第 6 站重新加工。当操作员把部件放到一个空的载体上面时，按下 PB2，输出指示灯 O:4/6 变亮。

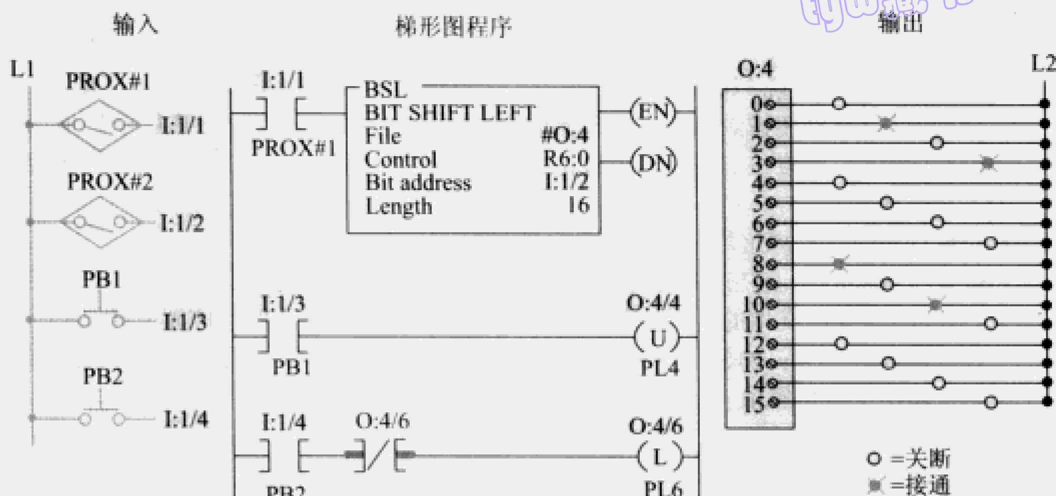


图 12-27 用位移位指令实现跟踪通过 16 站机器的载体

12.5 字移位寄存器

位移位寄存器被称为同步寄存器，因为在一个字内部或字之间，信息一次只移动一位。同步移位寄存器也被叫做串行移位寄存器。

异步字移位寄存器允许数据在文件中堆放。需要两个独立的移位脉冲：一个移动数据到文件（装载），另一个把数据移出文件（卸载）。两个移位脉冲的操作相互独立（异步）。PLC 的字移位寄存器有两种基本类型：

- ❑ FIFO（先进先出）
- ❑ LIFO（后进先出）

图 12-28 所示为一个 FIFO 指令对的例子。这两种 FIFO 指令都属于输出指令，并且成对使用。FIFO 装载（FFL）指令把数据从源地址单元载入文件；FIFO 卸载（FFU）指令把数据从文件卸载到目标字中。当成对使用时，这两个指令设置一个异步移位寄存器（堆栈），按照规定的次序存储和接收数据。 345

对 FIFO 堆栈进行编程时，需要对成对的两个指令设置相同的文件、控制地址、长度和位置值。需要给处理器提供下列信息：

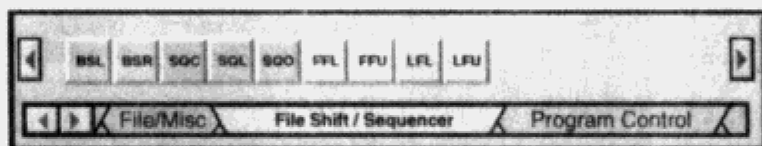
- ❑ **源地址**（Source）是输入到 FIFO 文件的数据来源地址。装载指令从该地址接收数据，并载入到堆栈的下一个字。
- ❑ **目的地址**（Destination）是存储从堆栈中弹出数据的地址。它是 FFU 指令上升沿脉冲到来时，数据在 FIFO 文件中变址的目的地址。当 FFU 变址时，目的地址中的数据被新的数据覆盖。
- ❑ **FIFO** 是堆栈的地址。它必须是输入、输出、状态、位或整数文件中变址的字地址。编程时 FFL 指令和 FFU 指令使用相同的地址。
- ❑ **控制**（Control）是控制结构的文件地址。状态位、堆栈长度和位置数据都存储于该单元中。FIFO 装载和卸载共享同一个控制单元，这个控制单元不能用于控制其他的指令。FIFO 控制单元的控制位如下：

位：	15	14	13	12
	EN	EU	DN	EM

346

EN 为 FIFO 装载指令的使能位，它和 FFL 指令的状态一致。EU 为 FIFO 卸载指令的使能位，它和 FFU 指令的状态一致。DN（完成位）指示位置值已经达到 FIFO 的长度，也就是说 FIFO 文件已经满了。当 DN 位被置位，它阻止任何数据从源地址到 FIFO 文件传输。当最后一段数据从源地址输入到目的地址并且位置值为 0，EM（空位）被置位。在 EM 位置位后，当 FFU 指令有上升沿脉冲到来时，0 被载入目的地址。

- **长度**（Length）规定堆栈中字的最大数量。
- **位置**（Position）是在 FIFO 文件中的指针。它指示了下一段数据存放的地址和已经存放到 FIFO 文件的数据段数。如果指令在上电时从一个偏移位置开始执行，输入该位置值即可；如果不是从偏移位置开始，则直接输入 0。根据需要可以在梯形图程序中更改位置值。



命　　令	名　　称	说　　明
FFL和FFU	FIFO装载和FIFO卸载指令	FIFO指令载入字到文件中，又按照载入的顺序把它们卸载出文件，先载入的字也先卸载出

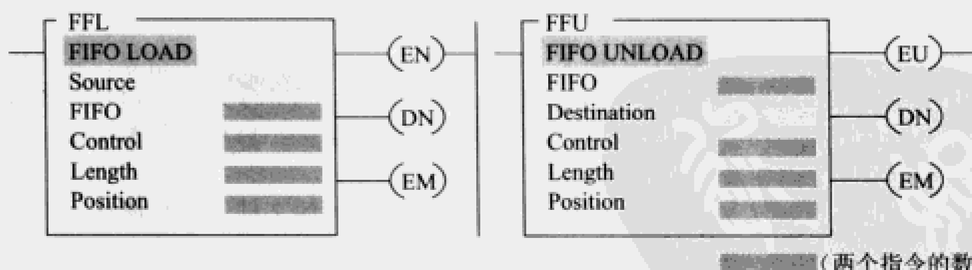
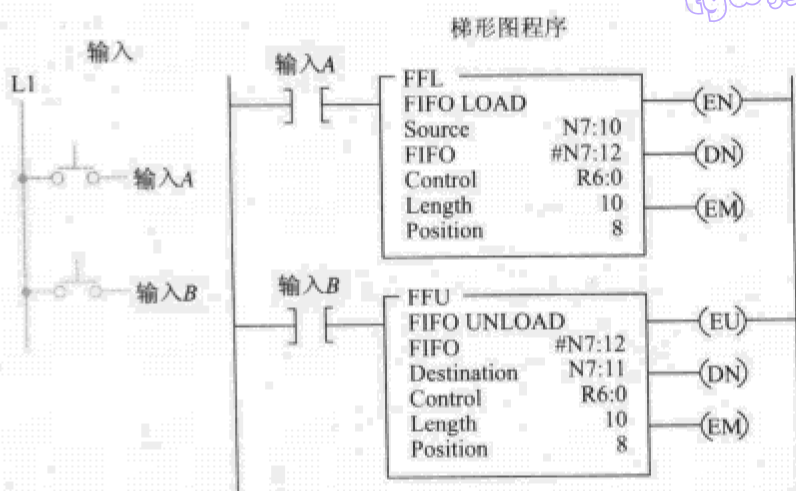


图 12-28　FIFO 装载（FFL）和 FIFO 卸载（FFU）指令对

图 12-29 所示的程序可以用来说明如何使用 FFL 指令和 FFU 指令对实现数据转进转出 FIFO 文件。当输入 A 的上升沿到来时，地址 N7:10 中的数据被输入到 FIFO 文件。在 FFL 指令的上升沿，数据被放于指令中指示的位置，执行后位置值指示当前进入 FIFO 文件的数据数量。在输入 A 的每一次上升沿，FIFO 文件输入从开始地址变更为高一个地址。输入 B 的上升沿引起 FIFO 文件中的所有数据向文件开始地址移动一个位置，开始地址的数据移动到目的地址 N7:11。

347

FIFO 指令常常用于存储控制。例如，有一定数量的元件要生产，每个元件被分配了不同的号码。一旦 PLC 控制生产开始，不同编码的元件可以采用 FIFO 指令，按照规定的次序拖出。



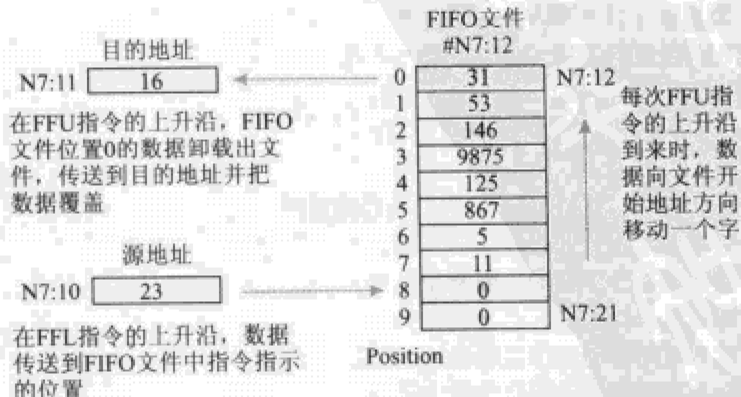
Integer Table

	Value
N7:10	23
N7:11	16
N7:12	31
N7:13	53
N7:14	146
N7:15	9875
N7:16	125
N7:17	867
N7:18	5
N7:19	11
N7:20	0
N7:21	0

Radix: Decimal　Table: N7: Integer　Forces

Address　Symbol

(a) 程序

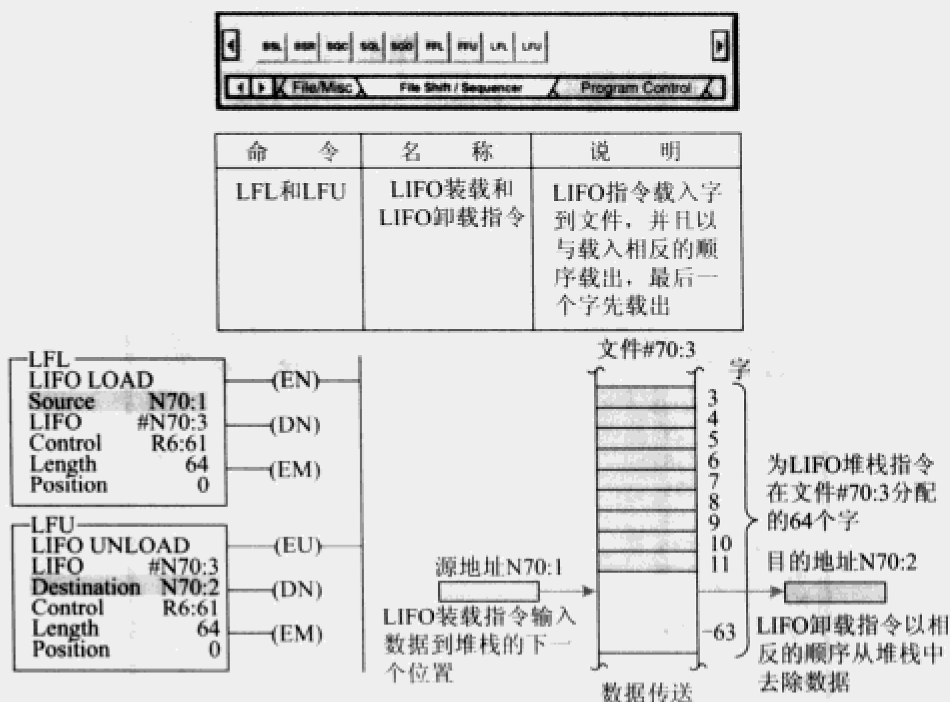


(b) 数据传送

图 12-29　数据在 FIFO 文件中转进和转出

LIFO 指令（后进先出）把接收数据的顺序颠倒，最先接收的数据最后输出，最后接收的数据最先输出。LIFO 本质上是一个堆栈，它允许数据加入，但不打乱原先已经包含在堆栈中的数据。如图 12-30 所示为一个 LIFO 指令对的例子。FIFO 和 LIFO 堆栈指令的区别是 LIFO 指令去除数据的顺序与它载入数据的顺序相反（后进先出），其他的都与 FIFO 指令相同。

348~349



命　令	名　称	说　明
LFL和LFU	LIFO装载和LIFO卸载指令	LIFO指令载入字到文件，并且以与载入相反的顺序载出，最后一个字先载出

图 12-30　LIFO 指令对

思考题

1. 解释凸轮顺序器开关的基本操作原理。
2. 顺序器开关最适合于什么类型的操作？
3. 顺序器编程与传统的编程方法相比优势在哪里？
4. 参考 PLC 顺序器输出指令，回答下列问题：
   a. 每一个顺序器步的信息输入到什么地方？
   b. 输出字的功能是什么？
   c. 解释顺序器指令在各步中前进时数据的传输过程。
5. 解释屏蔽字与顺序器指令联合使用的目的。
6. 在一些顺序器输出指令中的两个限制是什么？
7. 顺序器指令通常是保持型的，意义何在？
8. 解释事件驱动和时间驱动顺序器指令的区别。
9. 解释顺序器输入和顺序器比较指令的功能。
10. 解释顺序器载入指令的功能。
11. 位移位寄存器怎样处理单独的位？
12. 列出移位寄存器的四种常见应用。

13. 说出两种类型的位移位指令。
14. 说出用于异步字移位寄存器指令的两种类型的移位脉冲，并陈述各自的功能。
15. 解释 FIFO 寄存器和 LIFO 寄存器的区别。

350

习题

1. 参考图 12-5 的洗碟机电路（时间步间隔 45s），回答以下问题：
   a. 定时器里面有多少个凸轮开关？
   b. 一个完整的操作周期内有多少个时间步？
   c. 每一步的时间间隔是多少？
   d. 陈述定时器控制的五种输出设备。
   e. 一个完整周期内加热器接通的总时间长度是多少？
   f. 在周期中的第 20 分钟时，哪些输出设备接通？
   g. 填充阀保持接通的最长时间是多少？
   h. 解释安全水位开关的功能。
   i. 说出第一次清洗部分的各输出设备的动作次序。
   j. 为什么在整个操作周期定时器电机只在一个步中关断？
2. 构建一个与图 12-31 所示的六步凸轮顺序器相当的顺序器数据表。

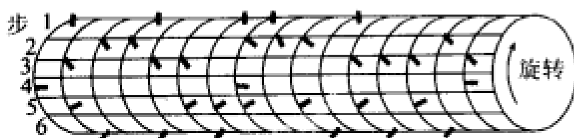


图　12-31

3. 参考图 12-32 的顺序器字文件回答下列问题：
   a. 假设输出位地址 1 到 16 控制了 1 到 16 盏灯，说出每一步中各个灯的状态。
   b. 哪些输出位地址应被屏蔽？
   c. 说出顺序器周期第三步中，输出字 25 中各个位的状态。
   d. 如果字 31 采用 16 进制数输入 PLC，应该怎么写？

	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	
字25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	输出
字30	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	第一步
字31	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	第二步
字32	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	第三步
字33	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	第四步

图　12-32

351

4. 参考图 12-12 所示的时间驱动顺序器交通灯程序回答以下问题：
   a. 顺序器控制了多少位的输出？
   b. 控制输出的位地址是什么？
   c. 设置输出状态的顺序器文件的地址是什么？
   d. 存储定时器当前值的顺序器文件的地址是什么？

e. 红灯接通编程设定时间是多少?
f. 绿灯接通编程设定时间是多少?
g. 黄灯接通编程设定时间是多少?
h. 一个完整的顺序器周期所需的时间是多少?
i. 假设存储在 N7:13 中的数据变为 35，说出数据交通灯的工作过程的变化。

5. 参考图 12-14 所示的事件驱动顺序器输出指令回答下列问题:
a. 顺序器何时步入下一步?
b. 如果顺序器在图所示的位置 2，哪些输出位接通?
c. 假设顺序器步入位置 8，哪些输出位接通?
d. 假设顺序器在位置 8，并且输入 A 的上升沿到来，会发生什么情况?

6. 采用熟悉的 PLC 顺序器输出指令，设计一段程序使气缸按照规定的次序动作。每步之间的间隔为 3s。期望的动作顺序如下:
- ❑ 所有的气缸收缩。
- ❑ 气缸 1 扩张。
- ❑ 气缸 1 收缩，气缸 3 扩张。
- ❑ 气缸 2 扩张，气缸 5 扩张。
- ❑ 气缸 4 扩张，气缸 2 收缩。
- ❑ 气缸 3 收缩，气缸 5 收缩。
- ❑ 气缸 6 扩张，气缸 4 收缩。
- ❑ 气缸 6 收缩。
- ❑ 重复循环。

7. 采用最熟悉的 PLC 顺序器输出指令，设计一段程序，实现汽车自动清洗过程。过程由汽车事件驱动，当汽车通过传动带链在汽车清洗机架上面拖动时，会碰到一些限位开关（LS1 ~LS6）。汽车清洗步骤如下:
- ❑ 汽车连接到传送带，并放入汽车清洗机架。
- ❑ LS1 打开输入水阀门。
- ❑ LS2 打开肥皂释放阀门，与输入水阀一起提供冲洗水雾。
- ❑ LS3 关断肥皂阀门，输入水阀门仍然在清洗车辆。
- ❑ LS4 关闭输入水阀门，打开热蜡阀门（如果选择）。
- ❑ LS5 关闭热蜡阀门，开启鼓风机。
- ❑ LS6 关闭鼓风机，汽车完成清洗。

352

8. 采用最熟悉的 PLC 顺序器输入和输出指令，设计一段程序，包含输入和输出指令对，实现以下要求:

输入为真使输出在指示的输出步内接通	输出步	在指示的输出步内输出为真
I:002/00、I:002/10	1	O:015/15、O:015/17
I:002/11、I:002/15	2	O:015/04
I:002/11	3	O:015/03、O:015/13
I:002/05、I:002/07	4	O:015/10
I:002/04	5	O:015/11、O:015/16

屏蔽掉所有没有使用的输入和输出位。构建一个表，列出需要输入到顺序器输入和顺序器输出文件的数据。

9. 产品连续通过一个有四个站的装配生产线，如图 12-33 所示。产品进入检查区，通过接近开关检测它是否存在。产品检查员对它进行检查，如果不合格就会按下报废按钮。报废状态指示灯在站 1、站 2 和站 3 变亮，告诉装配人员忽略此产品。当不合格的产品到达站 4

时，一个转换门被激活阻止该产品，使它到报废箱。使用最熟悉的 PLC 位移位寄存器，设计一段程序实现这个过程。

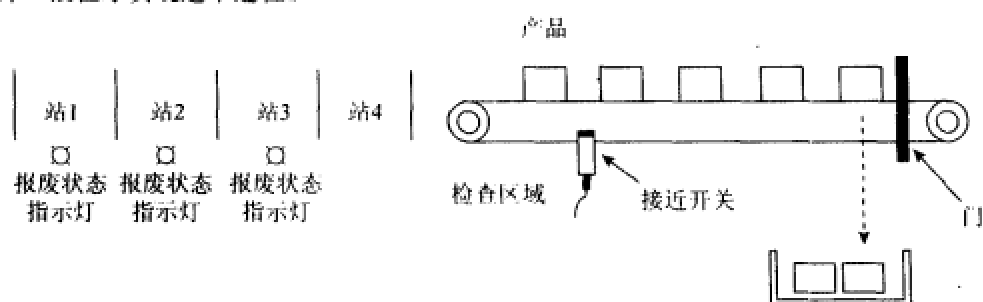


图　12-33

第13章　PLC安装、程序编辑和故障诊断

学习目标：

- ❑ 描述PLC的防护外罩应满足的条件。
- ❑ 能够辨识PLC的分泄电阻并描述它的功能。
- ❑ 能够区分离线情况和在线情况下的编程。
- ❑ 能够正确的表述PLC系统的接地措施和预防性维护工作。
- ❑ 能够描述和列写出PLC故障诊断的详细步骤。

本章讨论了PLC控制系统的安装、维护和故障诊断的标准要求，以及如何正确的接地来保证人身安全和设备的正常运行。文中还对PLC独特的故障诊断步骤做了说明。

13.1　PLC的防护外罩

正确安装的PLC控制系统，运行几年都不会出现故障。PLC本身的设计就包含了许多抗干扰的特征，这使得它几乎可以在任何工业环境下安装使用。但是如果系统安装得不够正确，还是会出现一些问题。

PLC需要保护措施来对抗极限温度、潮湿、灰尘、冲击、振动和腐蚀性环境。基于这些原因，PLC通常被安装于机器内部或者安装在一个单独的防护外罩内（如图13-1所示）。防护外罩是与外面大气环境隔离的最主要的保护措施。美国国家电气制造业协会（NEMA）根据提供的防护等级，把防护外罩进行了分类。对于大多数的固态控制装置，推荐使用NEMA 12型的防护外罩。这种防护外罩可用在一般的使用场合，并能有效的隔离灰尘。而且，金属防护外罩可以降低周围设备产生的电磁辐射影响。防护外罩并不能防止内部的冷凝，冷凝通常由温度的波动产生。为了防止冷凝现象和极限温度，可以考虑在防护罩内部安装一些加热设备。

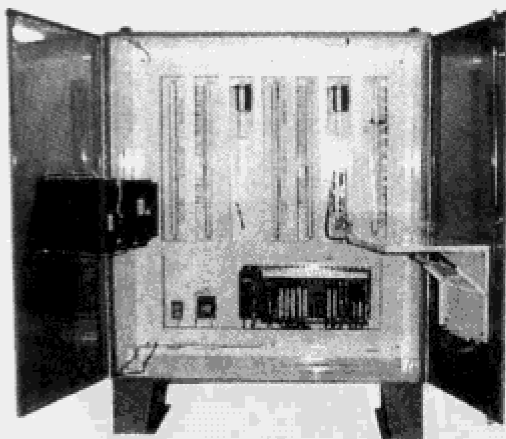


图13-1　安装于防护外罩内的PLC系统
（得到Industrial Solutions公司许可）

PLC系统应能把由电源、本地I/O模块和CPU产生的热量驱散出去。这些热量容易在防护罩内累积，所以必须把它散失到周围的空气中。对于许多应用场合，一般的对流冷却就能将控制器温度保持在规定的运行范围内。防护罩内的设备保持一定的间距，通常就能保证足够的散热。防护罩内的温度一定不能超过控制器的最大运行温度值（通常为60℃）。当遇到高温环境时，可能还需要预备额外的冷却设备，如风扇、鼓风机等。图13-2显示的是PLC系统安装的典型布局。

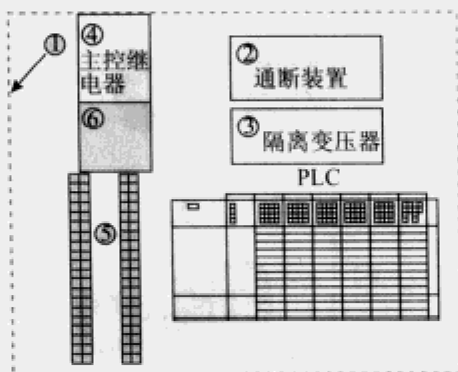
防护外罩应安装电源通断开关，因为有些情况下，PLC需要在断开外部电源时工作。还要设计一个观察口，这样就可以在正常运行期间看到仪表、PLC及其模块，而不必把防

护外罩打开。观察口被安装于控制面板的门上，这样方便工厂人员进行日常维护和故障检查。

防护外罩通常还配备了硬接线的主控继电器（MCR），当系统出现故障时它可以中断I/O模块的电源，但仍然保持CPU的电源。而且，隔离变压器提供以下的功能：

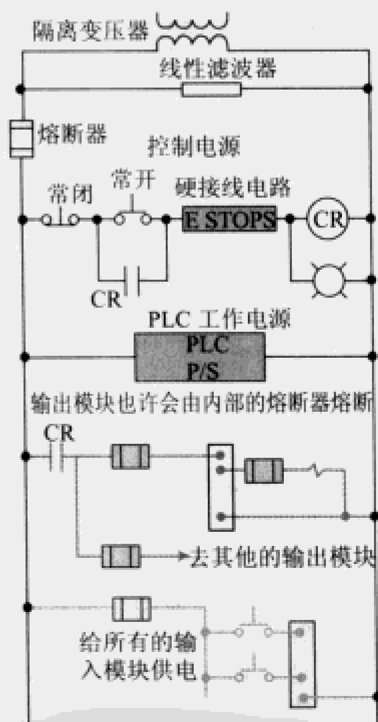
- 与主配电源进行物理隔离。
- 在需要的情况下改变电压，可以向交流配电系统提供110V或240V的电压。

355 ~ 356



① 适合应用条件和外部环境的NEMA标准的防护外罩，保护控制器免受电磁噪声和空气污染的影响
② 通断装置，把电源从系统中去掉
③ 根据系统需要选择的熔断型隔离变压器或恒压变压器
④ 主控继电器/急停电路
⑤ 接线盒或接线排
⑥ 电磁干扰抑制装置

(a) 典型的元件布置图



(b) 典型的接线方式

图 13-2 典型的 PLC 系统安装

13.2 电噪声

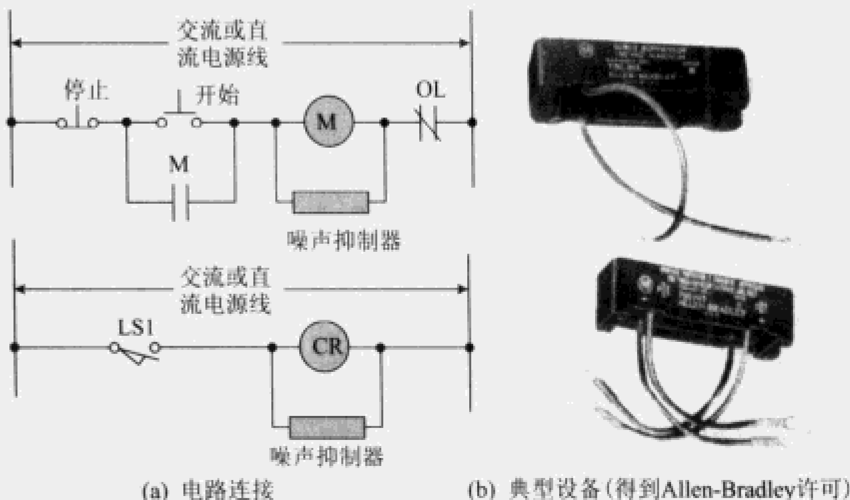
电噪声又被称为电磁干扰（EMI），是有害的电信号，能产生不期望的影响，并扰乱控制系统电路。电磁干扰（EMI）是辐射的或是传导的。辐射噪声由噪声源产生，并通过空气传播。收音机和电视信号就属于辐射的电磁干扰。传导噪声在实际的导体中传播，如电源线。原始噪声也许先是辐射传播，然后耦合到线路上，继而进行传导传输。

当PLC运行于噪声污染的工业环境时，要特别考虑可能的电磁干扰影响。由噪声引起短暂的运行偏差故障，在一些特定的应用场合就能导致危险的机器故障。噪声通常是通过输入端、输出端和电源线进入系统。而噪声可以通过静电场或电磁干扰耦合到这些线路上。下面的几个方面能减少电磁干扰的影响：

- PLC本身的结构特点
- PLC在防护罩内的正确安装
- 可靠的设备接地
- 正确的布线

□ 对产生噪声设备的可靠抑制措施

为了增加系统抗噪声的裕度，控制器的安装要远离能产生噪声的，如大功率交流电机和高频焊接机等的设备。潜在的产生噪声的设备包括：继电器、电磁阀、电动机和电机启动器。由其是当通过像按钮或选择开关这样的硬触点进行控制的时候，产生的噪声更加严重。当这些类型的设备作为输出负载，或与 PLC 采用同一电压供电时，对噪声的产生采取抑制措施是非常有必要的。图 13-3 显示了典型的噪声抑制方法。



(a) 电路连接　　(b) 典型设备(得到Allen-Bradley许可)

图 13-3　典型的噪声抑制方法

如果对感性负载浪涌抑制不够，就会造成处理器故障和运行不正常，随机存储器（RAM）数据丢失，I/O 模块也会有故障或者自复位。在感性负载被激励或者停止激励的瞬间，会产生一个返回 PLC 系统的电脉冲信号。当这个脉冲信号进入到 PLC 系统时，PLC 会把它误认为是正常的工作信号，就会打乱 PLC 正常的运行次序。这个单独的干扰脉冲就会产生一个 PLC 操作顺序流的故障。因此在噪声很严重的情况下，采用可编程的存储器模块自动调用处理器故障程序，或者使电源重新上电使系统快速恢复。

357

合理的布线也可以降低电磁噪声。在防护外罩的内部，供给处理器模块的输入电源线和供给 I/O 模块的导线要分开布线。决不能把信号线和动力线安装于同一个管道内。根据信号的类型把连接 I/O 模块的导线分开，具有相同信号特征的导线捆绑在一起，具有不同信号特征的线，要通过分开的路径进入到防护外罩内。对于光纤系统，虽然各种电磁干扰对它不会产生影响，还是要根据信号进行布线。

13.3　泄露输入和输出

很多应用于 PLC 控制系统的现场输入设备，如接近开关，都是固态构造。任何一种用固态开关硅控整流器（SCR）、双向可控硅或者是晶体管构成的电信号输入传感器，都有小的漏电流存在，即使在断开状态也不例外。通常这样的漏电流，只能引起模块输入端仪表指示的闪变。但是对于 PLC 系统的输入，这样的信号就可能导致假象的有效输入信号。为了解决这个问题，如图 13-4 所示，可以把一个分泄电阻跨接或是并联于输入端。

358

用于许多输出模块的固态开关也将会产生泄漏。把高频输出负载设备连接于这些输出模块，也会产生类似的问题。图 13-5 显示了如何连接分泄电阻，来除掉不期望的漏电流。

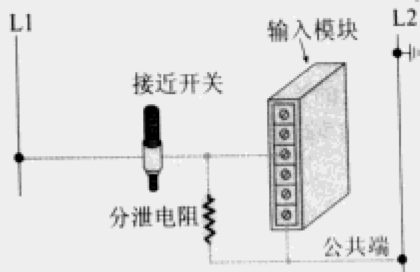


图 13-4 泄漏输入设备的连接

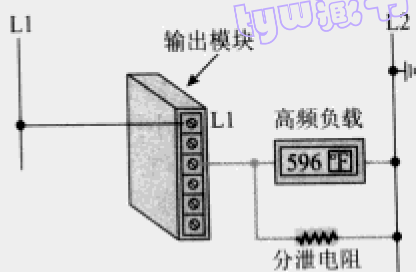
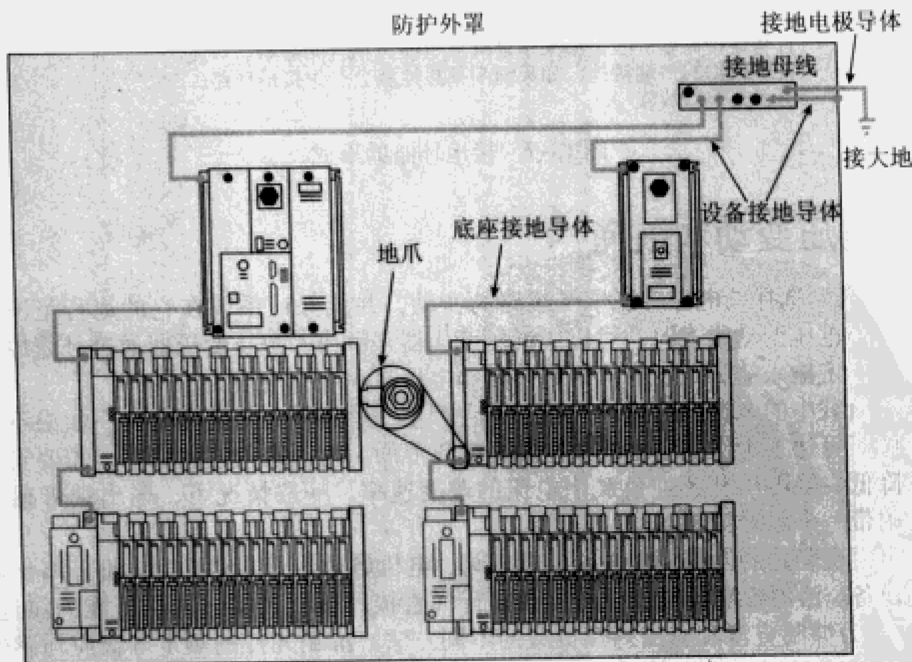


图 13-5 泄漏输出设备的连接

13.4 接地

正确的接地是所有电气安装中一个重要的安全措施。PLC 系统安装接地的权威标准是美国国家电气规程。规程中详细规定了接地导体的类型，颜色码和电气部件安全接地的连接措施。规程还规定了接地路径必须是牢固的（无焊接）、连续的，并且具有最小的接地电阻，能够安全的传导接地故障电流。在接地电流比较大的时候，导体的温度升高容易把焊接剂熔化，造成接地连接的中断。除了控制器和它的防护外壳要正确接地外，其他的一些应用设备也要正确接地。大多数的生产厂家，都提供使用防护外壳时的详细接地方法。图 13-6 所示为防护外罩的典型接地连接。

359



注意：当采用这种接地方式时，在电源的终端不要设备接地，否则会引起接地环路

图 13-6 PLC 的接地连接

对于固态控制系统，接地能够限制电磁感应引起的噪声影响。以下的接地措施能够减少电磁干扰：

- 所有的 PLC 设备和外壳背板应该单独接地，并连接到防护外罩的支架中心点。

- 进入防护外罩的电源线和接地线应该分开。
- 在接地线、地爪、和金属防护外罩表面间的接地连接都要用星形垫圈。
- 油漆和其他的非导体材料应远离底座和防护外壳的连接处。
- 对于 PLC 设备，最小接地导线应为 AWG 标准的 12 号铜线。对于外壳背板，最小接地总线应为 AWG 标准的 8 号铜线。
- 防护外罩应与接地母线正确连接。
- 机器应与防护外罩连接并与大地连接。
- 接地连接应有非常小的电阻。按经验，设备和地之间的电阻应该小于 0.1Ω。

接地环路能引起噪声问题，而且不易发现，当出现多点接地时就会发生。接地点离得越远，越容易产生噪声问题。在电源地和远方地之间也有产生接地环路的可能。多点接地形成拾取干扰的通路，造成接地环路干扰。当变磁场区域通过环路时，就会在回路中产生电压和电流。接地环路形成图示如图 13-7 所示。

360

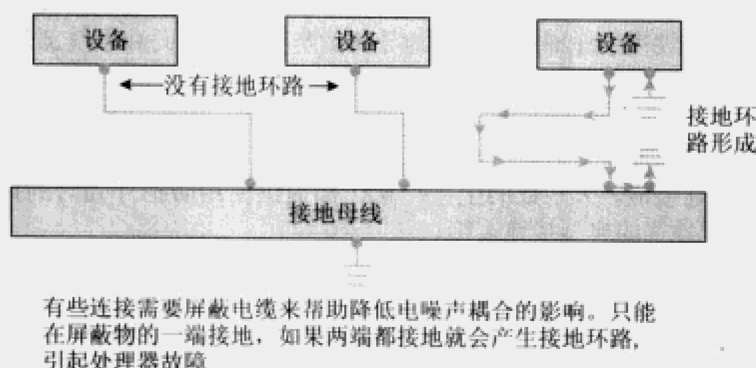


图 13-7　接地环路的形成

13.5　电压变动和浪涌

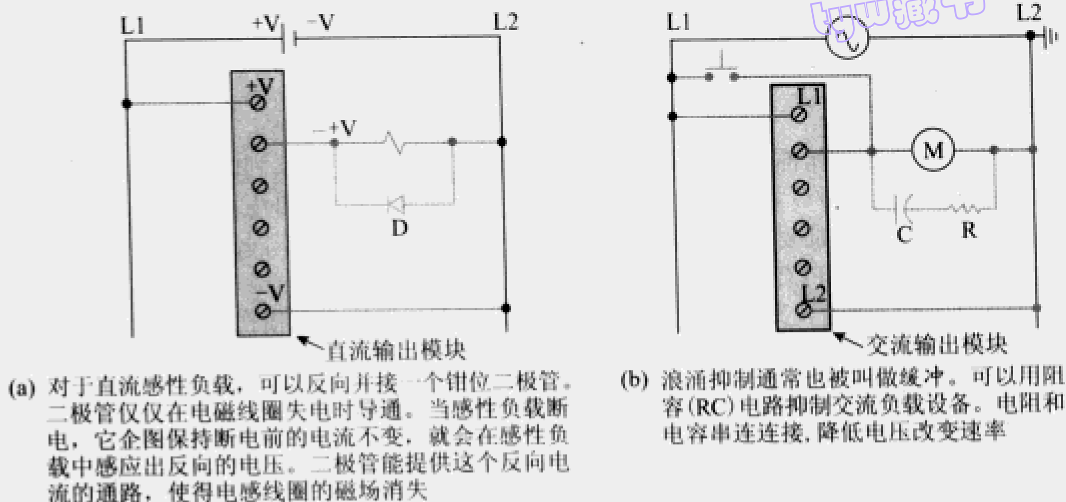
为 PLC 系统选择的电源应能承受线路的变动，并能保证系统在它的运行范围内正常工作。在线路电压变动很大时，可以用恒压变压器来解决。恒压变压器通过补偿原边电压的改变值，稳定输入端的电压，从而维持副端电压在一个稳定值。

当感性负载中的电流中断或消失时，就会产生一个非常大的尖峰电压。如果不进行抑制，尖峰电压可达几千伏并产生破坏性的高电流浪涌。为了避免这种情形，需要安装抑制电路，来降低尖峰电压和感性负载中电流的改变速率。一般情况下，输出端带感性负载时，都要附带一个浪涌抑制电路。

当输出模块驱动的设备是继电器、电磁阀、电机启动器或电动机时，推荐增加额外的浪涌抑制设备。浪涌抑制设备应该与负载并行（直接跨接）连接，并尽可能的靠近负载。图 13-8 所示为抑制直流和交流感性负载的不同方法。抑制元件的额定等级应与感性负载的开关瞬态特性相匹配。查阅 PLC 系统的安装手册，选择正确等级和类型的抑制设备。

361

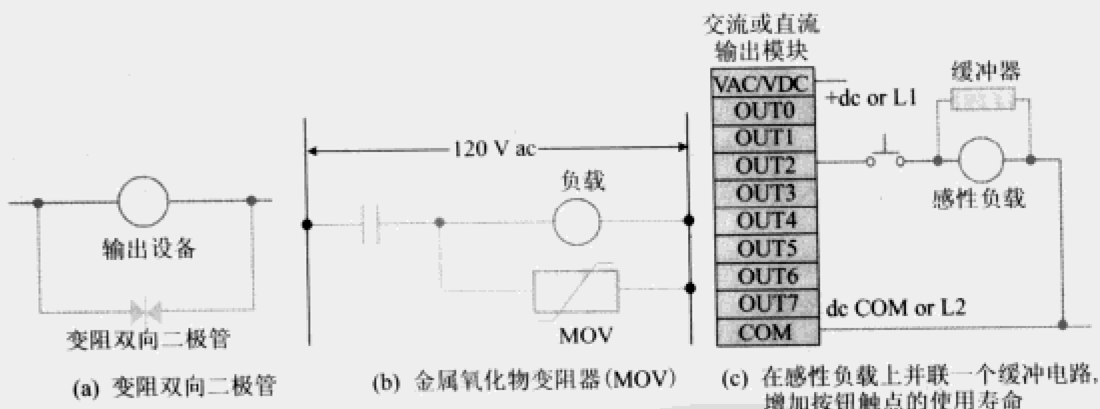
金属氧化物变阻器（MOV）浪涌抑制器（如图 13-9 所示）在功能上与背对背齐纳二极管相当。在没有反向电压加在齐纳二级管上并超过它的额定电压时，齐纳二极管都是常开的。任意一个超出额定值的尖峰电压都会使齐纳二极管像短路一样，这样就可以把这个电压从剩余的电路中去掉。如果感性负载串联或者并联硬触点，如按钮或选择开关，增加浪涌抑制是非常重要的。在没有浪涌抑制的情况下，开断感性负载会大大降低触点的使用寿命。安装抑制设备时要与负载设备尽可能地靠近。



(a) 对于直流感性负载，可以反向并接一个钳位二极管。二极管仅仅在电磁线圈失电时导通。当感性负载断电，它企图保持断电前的电流不变，就会在感性负载中感应出反向的电压。二极管能提供这个反向电流的通路，使得电感线圈的磁场消失

(b) 浪涌抑制通常也被叫做缓冲。可以用阻容(RC)电路抑制交流负载设备。电阻和电容串连连接，降低电压改变速率

图 13-8 直流和交流感性负载的抑制



(a) 变阻双向二极管

(b) 金属氧化物变阻器(MOV)

(c) 在感性负载上并联一个缓冲电路，增加按钮触点的使用寿命

图 13-9 缓冲电路

13.6 程序编辑

当输入梯级程序后，还要对它进行修改。编辑就是通过一系列的编辑功能对已经存在的程序进行修改。利用编辑功能，可以增加、删除指令和梯级，还可以改变地址、数据和位的状态。当然，不同的生产厂家和机型有不同的程序编辑模式。

现今，大多数的 PLC 编程软件都是基于微软视窗的，所以如果熟悉微软系统并懂得如何使用鼠标，那么编辑程序是毫无问题的。一般来说，选择指令和梯级时，可以很简单地用鼠标左键在上面单击一下。双击鼠标左键可以对指令的地址进行编辑，而单击鼠标右键可以弹出涉及编辑指令的下拉菜单。如果想要符号或地址包括额外的解释，可以直接在梯级相应的符号上面增加地址解释性文字；要增加页或梯级注解，只需在相应的梯级上单击鼠标右键。

362

当编辑梯形图逻辑时，查找功能非常有用。它可以用来查找那些包含特定地址指令的程序。启动查找功能可以自动定位存储器中特定地址的指令。包含被查找指令的电路就会自动显示在屏幕上，供用户检查。如果需要，用户可以对指令，或是包含指令的电路进行修改。根据不同的 PLC，查找功能可以用来查找：

- 带地址的指令
- 指令类型
- 地址
- 程序梯级

一个控制系统开始准备运行，通常被称作试机，它包括了一系列的测试试验，来确保PLC、梯形图程序、I/O 模块和所有的相关接线，都能符合工序和技术要求。在任意一个控制系统试机之前，都要充分了解控制系统是如何工作的，以及不同部件之间是怎样相互影响的。下面就是对 PLC 系统试机的一般步骤：

(1) 在 PLC 或输入设备上电前，要去除或隔离那些可能产生损伤或破坏的输出设备。这种预防措施适合能引起移动的输出，如启动电机或是开启阀门。

(2) PLC 和输入设备上电，并测量电压是否在额定值。

(3) 检查 PLC 的状态指示灯。如果电源正常，指示灯会亮并且没有故障指示。如果上电不正确，它可能有故障。PLC 不容易出故障，如果出现故障，通常也是在电源上电的瞬间。

(4) 确认 PLC 与手持式编程器或是装有编程软件的个人计算机通信连接好。

(5) 把 PLC 放在输出电路不能被激励的工作模式。根据 PLC 的不同结构，这种模式可以被叫做：*不能执行*，*连续测试*，或*单次扫描模式*。在这种模式下，可以监视输入设备、执行程序和在保持输出不能被激励的情况下更新输出映像寄存表。

(6) 手动一次一个的激活输入设备，确认 PLC 输入端的状态信号指示灯显示正确的通断状态。监视与之相关的状态指令，确定输入与程序地址对应正确，并且指令按既定期望动作。

(7) 手动测试每个输出。可以采用与输出设备相连的终端通电的方法，这个测试可以检查外部现场设备及其接线。

(8) 检验了所有的输入、输出和程序地址后，再检验所有的计数器、定时器等的预置值。

(9) 把没有连接好的输出设备重新接好，把 PLC 置于运行状态，检验所有的急停按钮和整个系统的运行状况。

13.7 编程和监控

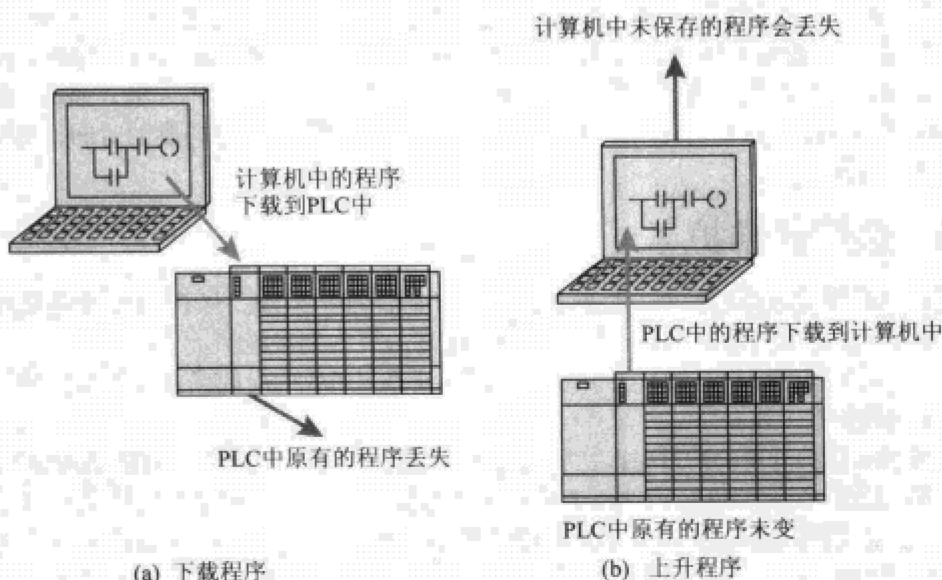
当对 PLC 进行编程时，根据不同的生产厂家和模块，有不同的指令输入模式。一般用装有编程软件的个人计算机来编写程序和监控存储于 PLC 里的程序。而且，还可以进行*离线编程*，即在计算机上编写和保存程序而不必把 PLC 与计算机连接，之后再把程序下载到 PLC。(如图 13-10 所示)。相反，*在线编程*就是把梯形图程序直接输入到 PLC。在离线情况下编程是一种最安全的方式，因为在没有把程序下载到 PLC 之前，增加、改变或是删除梯形图，都不会影响系统的运行。

363

很多厂家的 PLC 都提供一种*连续测试模式*，即处理器执行用户程序而不激励任何输出。在这种模式下，允许执行和调试控制程序而禁止输出。通过监控编程设备上相应的输出梯级，检查每个梯级是否被执行。在*单次扫描模式*下，也可以调试程序的控制逻辑。在单次扫描模式下，处理器一次只对用户程序扫描一次，而没有任何输出被激励。

在线编程模式，允许用户在系统运行期间，对程序进行修改。当 PLC 控制设备或进程时，用户可以根据需要增加、改变或删除控制指令和数据。任何被改变的程序，都会被立即执行。因此，用户必须事先评估出由程序改变带来的机器可能的运行次序。在线编程的操作员，必须是那些有经验的人，他们对 PLC 的运行和被控制的机器都非常了解。否

则，应该先进行离线编程，然后再用新程序安全地代替原来的程序。



(a) 下载程序　　(b) 上升程序

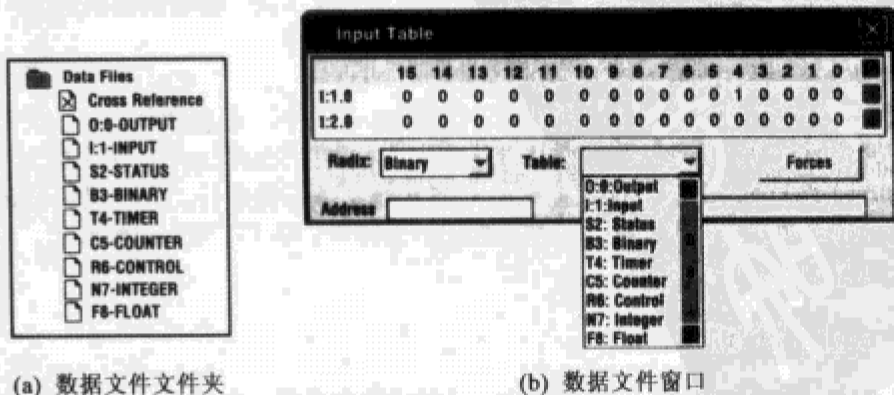
图 13-10

数据监控允许显示数据表中的任何数据。根据不同的 PLC，数据监控功能可以用于如下几个方面：

- 观察指令中的数据
- 存储指令用过的数据
- 在调试时，设置或重新设置数据值或位的状态
- 改变基数或数据形式

图 13-11 所示为 Allen-Bradley SCL-500PLC 和它的编程软件 RSLogix 的数据文件夹和窗口。数据文件夹允许用户决定文件的状态，这些文件包括：I/O 文件、状态文件（S2）、位文件（B3）、定时器文件（T4）、计数器文件（C5）、控制文件（R6）、整数文件（N7）和浮点数文件（F8）。用数据监控功能处理数据时，一定要小心，因为改变数据会影响到程序，并引起输出设备导通或关断。

364



(a) 数据文件文件夹　　(b) 数据文件窗口

图 13-11　Allen-Bradley SCL-500PLC 和它的编程软件 RSLogix 的数据文件文件夹和窗口

触点统计图（contact histogram）功能可以观察数据表中数据的变化记录（导通或关断状态），能显示位的状态（导通或关断）以及保持通或断的时间长度（以小时、分钟、秒和毫秒表示）。在触点统计图文件中，累积的时间是统计图功能运行的总时间，而delta时间表示的是状态改变间歇消耗的时间。无论是有关硬件的还是程序逻辑方面的，触点统计图对检测间断性问题非常有用。通过跟踪状态的改变和状态改变间歇的时间，可以检测不同类型的错误。

13.8 预防性维护

避免PLC系统故障的最好方法就是正确的预防性维护措施。尽管PLC本身就被设计为很少需要维护并能可靠运行，一些维护性措施还是要定期实施。

[365] 许多控制过程的控制系统，为了改变生产必须停运短暂的时间。在这段时间内，应该执行下面的维护性工作：

- 安装在防护外罩内的过滤器，应该清洁或更换，以保证防护罩内有清洁的空气流通。
- 必须清洁积聚在PLC电路板上的灰尘和污垢。如果灰尘积聚在散热器和电子线路上，会阻碍热量的扩散并导致电路工作不正常。更严重的是，如果导电的灰尘到达电路板，会引起短路故障，使电路板永久损坏。确保防护外罩的门关闭，以防止这些污染物的快速滋生。
- 检查与I/O模块的连接是否松动，确保所有的插头、插座、接线头和模块的连接完好和模块安装牢固。如果连接松动，不仅会引起控制器工作不正常，还会损坏系统的部件。
- 所有的现场I/O设备都要检查，以确保正常工作。处理过程控制模拟量的电路板，每隔6个月要校准一次，其他像传感器这样的设备，每月都要校准。系统的终端设备（把机械信号转换为电信号的设备），可能失灵、变脏、破裂或是损坏，或不能按正确的设置脱扣。
- 一定要注意，不要把能发出重噪声和产生热量的设备，和PLC靠的太近。
- 检查用来支撑CPU里RAM存储器的后备电池。大多数的CPU都有一个状态指示灯，能显示电池的电压是否还足够支撑存储于PLC里的存储器。如果要更换模块，一定要更换相同型号的。
- 一般情况下都需要空闲部件以满足存储需求。输入模块和输出模块是PLC中最容易出问题的组件。
- 保留一份所使用的操作程序的副本。

为避免人身伤亡和设备损害，所有的连接检查，都必须是在电源从系统去除后才能进行。在接通电源之前，其他的能量源（气压和水压）不能使系统运行。大多数公司，在进行维护和维修工作时，采取用锁或标记的方法确保设备不被运行。人身保护标记被放在PLC和设备的电源旁边，也只能由最初放标记的人移走。除了用放标记的方式，用锁锁住也可以达到避免设备运行的目的。

13.9 故障诊断

PLC系统一旦出现故障，要进行仔细、系统的故障诊断，来解决问题。PLC的故障诊断相对比较容易，因为控制程序可以在监视器上显示，并可以在它执行的过程中实时观测。如果控制系统已经投入运行，则可以完全确认程序逻辑的准确性；但是如果系统还没

有运行或者正在试运行，还是要考虑编程存在的错误。

当发生问题时，故障诊断措施的第一步，就是找出问题和故障源。故障源一般就局限在处理器模块、I/O硬件、接线、设备输入或输出，或者梯形图程序上面。一旦找出故障，解决起来就很简单了。以下内容介绍了对可能出现故障的区域进行故障诊断。 366

13.9.1 处理器模块

处理器能够对潜在的问题进行自检。在处理器运行期间可以进行错误检查，并发送检查状态信息到指示灯上面，指示灯通常固定在处理器模块的前面。典型的诊断包括：存储器正常、处理器正常、电池正常和电源正常。

处理器连续不断的自检，检查任何可能引起不能正常执行用户程序的故障。根据不同的控制器，可以采用一系列的故障继电器触点。故障继电器由处理器控制，并且当一个或几个特定故障连续发生时被接通，它的触点切断输出电路并发出故障信号。

大多数PLC都包含有一个看门狗定时器，监视系统的扫描过程。看门狗定时器通常是独立的定时电路，必须由处理器在预先设定的时间内对其置位和复位。如果处理器出现硬件故障，看门狗定时器超时，立即中断PLC的运行。例如，如果程序扫描周期和看门狗的设定值相等，看门狗电路就会指示故障。运行手册上有如何运用这项功能的说明。存储器中数据的错误，也可以通过一些嵌入的诊断程序检测。

PLC的处理器硬件一般不会出现故障，因为当今的微处理器和微型计算机硬件，在规定的温度、湿度等条件下，运行非常可靠。PLC处理器的底座也被设计成能适应恶劣的工业环境。

13.9.2 输入故障

如果PLC工作在运行模式，输出设备没有按照程序动作，最可能的故障源为以下几个：

- I/O设备
- I/O模块、I/O设备和用户电源之间的接线
- 电源
- I/O模块

通过比较可能存在问题的I/O的实际状态与控制器上面状态指示器，可以缩小故障源的范围。通常每一个输入和输出，至少有两个状态指示。一个是在I/O模块上面的状态指示灯，另一个在编程设备的监视器上。

如果怀疑输入硬件是故障源，先检查一下当输入模块接收来自输入设备（如按钮、限位开关等）的电信号时，对应的状态指示灯是否变亮。如果当输入设备接通，而输入模块对应的状态指示灯没有亮，就要对输入终端的电压进行测量，检查电压值是否正确（如图13-12所示）。如果电压正常，就要更换输入模块；如果电压不正常，就要检查提供给输入设备的电源，检查输入设备、输入模块和电源之间的接线和输入设备本身是否存在问题。 367

如果在编程设备监视器上，显示出的指令状态不正确，输入模块可能没有把输入信号转化为处理器能够接受的电压等级信号。这种情况下，需要更换输入模块。如果输入模块更换了以后，问题还没有解决，并且接线是正确的，那么问题可能存在于I/O机架、通信电缆或处理器上面。图13-13所示为典型的输入设备故障诊断指南。指南中回顾了状态指令以及它们的通/断状态跟随外部输入设备的变化情况。

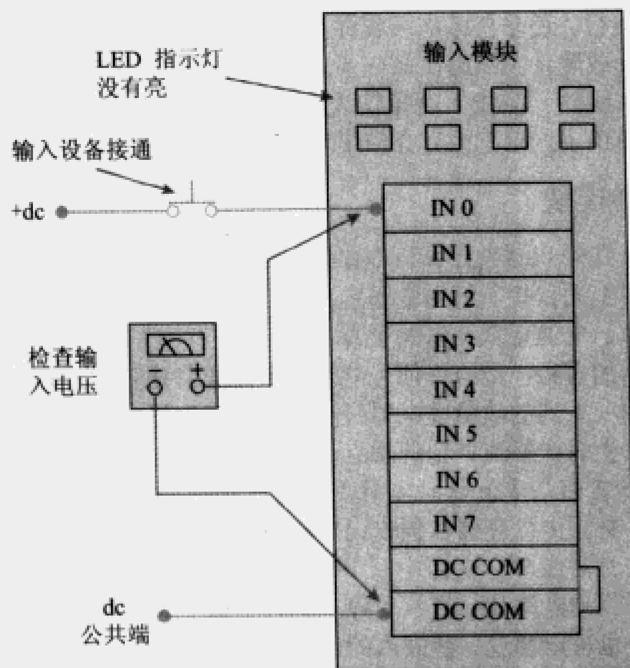


图 13-12 检查输入故障

输入设备故障诊断指南				
输入设备状态	输入模块状态指示灯	监视器显示的状态指示		可能的故障源
		⊣⊢	⊣/⊢	
闭合	亮	(1)	(0)	无故障，正确的状态指示
断开	灭	(0)	(1)	无故障，正确的状态指示
闭合	亮	(0)	(1)	1. I/O 模块 2. 处理器/操作终端通信
闭合	灭	(0)	(1)	1. 电源与I/O之间的接线 2. I/O 模块
断开	灭	(1)	(0)	1. 编程错误 2. 处理器/操作终端通信
断开	亮	(1)	(0)	1. 输入设备或线路短路 2. 输入模块

图 13-13 典型输入设备故障诊断指南

13.9.3 输出故障

如果输出不能按期望的被激励，首先应该检查输出模块上的熔断指示灯。通常只有当与熔断指示器对应的输出电路通电时，指示灯才会亮。如果指示灯是亮的，那么肯定就是保险丝的故障，需把保险丝更换掉。

如果熔断指示灯没有亮（保险丝是好的），就要检查输出设备和对应的 LED 状态指示灯是否正确。如果输出梯级通电，模块上面的状态指示灯变亮，但是输出设备没有反应，问题可能出现在输出设备的接线上或是输出模块本身。根据程序编程设备上的监控器，如果输出被控制通电，但是状态指示灯没有亮，那么需要更换输出模块。图 13-14 所示为典型的输出设备故障诊断指南。



输出设备故障诊断指南			
输出设备状态	输出模块上的状态指示灯	监控器上面的状态指示	可能的故障源
被激励	亮	(1)	无故障，状态指示正常
没有激励	灭	(0)	无故障，状态指示正常
没有激励	亮	(1)	1.与输出设备的接线 2.输出设备
没有激励	灭	(1)	1.输出模块上的熔断指示灯 2.输出模块故障

图 13-14 典型输出设备故障诊断指南

13.9.4 梯形图程序

如果程序曾经正确运行过，梯形图程序是不可能发生故障的。存储梯形图程序的 IC 存储器发生故障，就会改变程序的内容，但这也是 PLC 的硬件故障。如果其他的可能故障源都被排除了，那么需要把备份程序重新下载到 PLC 一次。

对程序进行故障诊断，首先要查明哪个输出工作正常，哪个输出工作不正常。然后，利用编程软件和查找功能，跟踪输出没有使能的梯级，检查梯形图逻辑，找出输出没有被激励的原因。通常的逻辑错误包括：

- 用常开触点代替了常闭触点（反之亦然）

❑ 在程序中用了不正确的地址

369

尽管梯形图程序可能没有出错，但是在最初编程的时候没有考虑周全，因此不能正确运行。这种情况下，需要对程序进行修改，而增加这个新的状态。通过对控制系统的过程描述和梯形图程序进行仔细的检查，可以查找这种类型的错误。

在调试时，强制闭合和强制断开指令允许特定位的状态置 1 或置 0。强制功能可以模拟运行或控制输出设备。例如强制一个电磁阀接通，看其是否在程序不起作用的情况下动作。如果是，问题肯定在软件上面，而不是在硬件方面。如果进行强制时，输出也没有反应，可能是输出模块存在问题或是阀门本身的故障所致。进行强制时，一定要做好充分的预防措施，保证人身和设备安全。

PLC 的指令系统中，包含了特定的故障诊断指令，用来进行故障检测。暂停（TND）指令（如图 13-15a 所示），可以改变程序扫描的范围，分段调试程序。当它的梯级条件为真时，处理器停止扫描 TND 指令以后的所有梯形图程序。当处理器扫描遇到 TND 指令时，复位看门狗定时器（清 0），更新 I/O，然后从主程序的第一个梯级重新扫描执行。

中止（SUS）指令（如图 13-15b 所示），用来在程序调试或系统故障诊断时，获得或查找具体条件。当它的梯级为真时，处理器处于挂起模式，运行被中止，并且中止 ID 号存放于状态文件的第 7 个字（S:7），这样就可以在梯形图中查找故障原因。包含了执行 SUS 指令的程序号，存放于状态文件的第 8 个字（S:8）。所有的输出都不被激励。

370

JMP LBL JSR RET SBR TND MCR SUS
Program Control Ascii Control Ascii String Micro

命　　令	名　　称	描　　述
TND	暂停	使执行程序暂停
SUS	中止	诊断程序调试和系统故障的具体情况

输入
TND

(a) TND指令

输入 A　输入 B　输入 C
SUS
SUSPEND
Suspend ID

(b) SUS指令

图　13-15

现场设备和 I/O 模块终端的接线，是容易发生故障的地方。错误接线和机械连接问题，都能使送入 I/O 模块，或由它们传出的信号中断或短接。

连接到过程输入/输出的传感器和执行机构，也可能出现故障。在运行期间，机械开关的使用寿命可能到限或损坏，电机、加热设备、灯和传感器也可能故障。为了保证正确的运行，输入和输出现场设备与输入/输出模块一定要兼容。

一些 PLC 生产厂家，允许相同的输出线圈在梯形图程序中多次使用。因此，就出现了多个梯级条件控制相同的输出线圈，增加了故障诊断时的难度。当出现双重线圈时，可能会出现通过监控器观察到梯级条件为真，而在梯形图控制相同输出线圈的梯级条件为假，PLC 还是保持输出断电。一些软件可以检测双重线圈的使用。

当问题发生时，采取的最好方法就是检查那些可能引起故障的地方，而不是武断地检查每一个连接、开关、电机、输入/输出模块等。首先，观察系统的运行并想法描述出问题的所在。通过对控制系统的观察和描述，查找可能的故障源。比较硬接线的输入、输出的逻辑状态和实际的状态（如图 13-16 所示）。如果有不一样，就会知道存在故障和故障存在的大体位置。 371

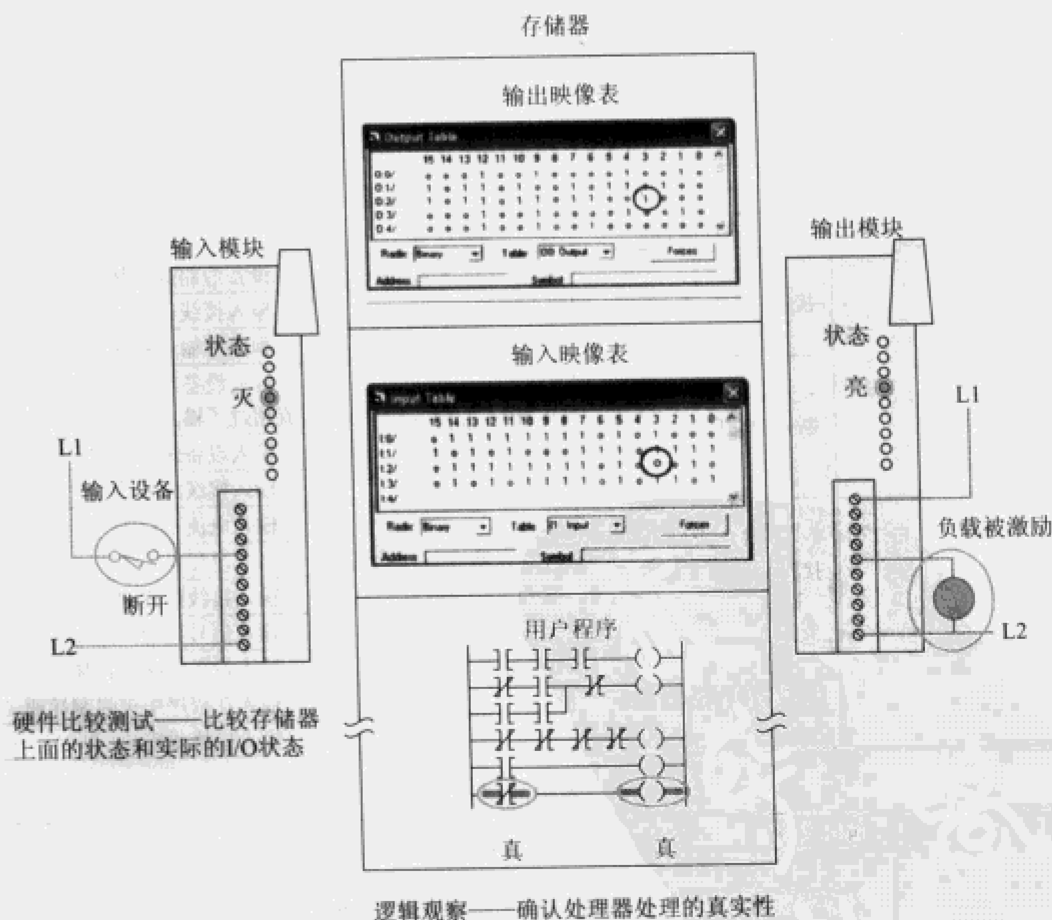


图 13-16 故障诊断的一般方法

大多数的故障诊断，都可以通过分析输入/输出模块上面的状态指示灯完成。关键是，能否通过状态指示灯得到存在错误或是系统正常。通常故障诊断的指南、图表或流程图上给出出现的故障以及它们可能的故障源。图 13-17 的故障诊断流程图和表 13-1、表 13-2 的 I/O 故障诊断指南都是这类典型的应用。

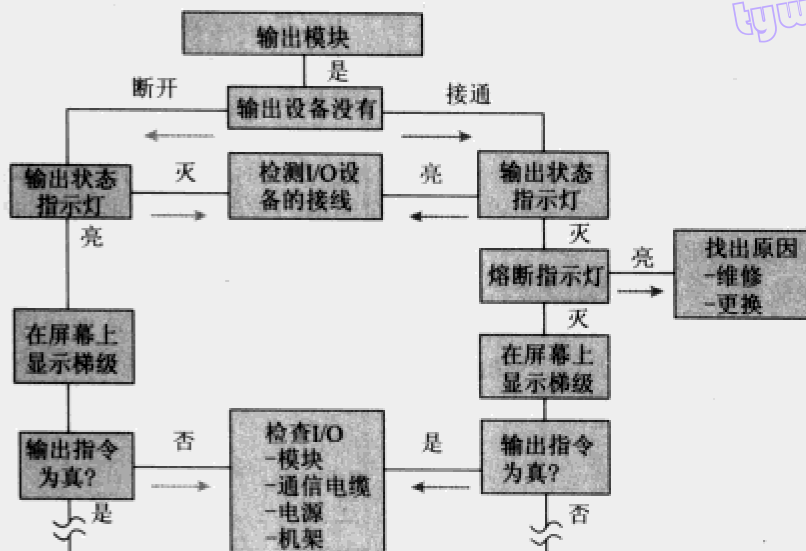


图 13-17 典型的故障诊断流程图

表 13-1 输入故障诊断指南

输入状态指示灯	输入设备	条件	可能的原因
亮	接通/闭合/工作	输入设备不能关断	设备短路或损坏
		程序以断开状态执行	输入接线或是模块
			程序中输入被强制关断
	断开/打开/没有工作	程序以闭合或输入电路不能关断状态执行	输入设备在断开状态的漏电流超过了输入电流的规定值
			输入设备短路或损坏
			输入接线或模块
灭	接通/闭合/工作	程序以断开状态执行；输入电路不能被接通	输入电流不匹配
			输入电压低
			输入接线或模块
			输入信号导通时间太短
	断开/打开/没有工作	输入设备不能接通	输入设备短路或损坏
		程序以闭合状态执行	输入在程序中被强制导通
			输入接线或模块

表 13-2 输出故障诊断指南

输出状态指示灯	输出设备	条件	可能的原因
亮	接通/被激励	程序显示输出电路断开或者输出电路不能被关断	编程原因： －检查双重输出和地址 －如果有子程序，当子程序没有执行时，输出不是它的最后状态 －用强制功能，把它强制关断。如果没有用强制关断功能，输出电路有故障；如果是强制关断的原因，就要再次检查程序存在的问题
			输出在程序中被强制接通
			输出接线或输出模块

（续）

输出状态指示灯	输出设备	条件	可能的原因
亮	关断/没有激励	输出设备没有运行，但是程序中指示它通电	负载电压低或没有电压
			输出设备不匹配：检查规格和源/漏的一致性（直流输出）
			输出接线和输出模块
灭	接通/被激励	输出设备不能被停止，而程序中指示的是它已经断电	输出设备不匹配
			输出设备断开状态漏电流超过了输出设备的规定值
			输出接线或模块
			输出设备短路或损坏
	关断/没有激励	程序中指示输出电路通电或输出电路不能接通	编程问题： －检查双重输出和地址 －如果有子程序，当子程序没有执行时，输出不是它的最后状态 －用强制功能，把它强制接通。如果没有用强制接通，输出电路有故障；如果是强制接通的原因，就要再次检查程序存在的问题
			输出在程序中被强制关断
			输出接线或输出模块

13.10　PLC与计算机的连接

必须设置一种方法，使个人计算机上的软件与PLC的处理器进行通信。设置这个连接通常被称为组态通信。不同品牌的PLC处理器有不同的通信组态方法。Allen-Bradley PLC用RSLogix软件输入和编辑梯形图程序，RSLinx软件包用来做以下工作：372

- 监控PLC的运行；
- 把程序从计算机下载到PLC；
- 把程序从PLC上传到计算机。

RSLinx根据价格和功能需求的不同，可以提供多种软件包。这个软件包被用做PLC和计算机通信的驱动。驱动就是用来控制设备的计算机程序。例如，必须在计算机上安装正确的打印机驱动，才能打印计算机上的字处理文档。RSLinx就相当于是RSLogix软件的打印机驱动程序。在计算机和PLC之间利用RSLogix软件建立通信连接之前，RSLinx程序必须先打开并把驱动设置好。如果需要组态驱动的帮助，可以通过RSLinx的帮助性文件，获得一步一步的指导。373

RSLinx允许RSLogix通过一条接口电缆与PLC的处理器进行通信。计算机与PLC之间最简单的通信连接方式是串行连接，因为它不需要任何的转换器和适配器。串行电缆连接计算机的COM1或COM2口和PLC处理器的串行端口。也许有人会认为任何带有RS-232串口的PLC处理器，都可以和任何其他带有RS-232串口的设备通信。其实并非如此，必须还要考虑通信链路的两个重要方面，即RS-232标准和通信协议。

RS-232标准，规定了标准通信电缆每根线和相应的引脚的功能，但是没有规定必须使用多少根线或引脚。双向通信的最小组态连接方式，只需要3根连接线，如图13-18所示。对于这种简易的连接方式，RS-232标准规定了计算机设备配备公插头，而外围设备配备母插头。两计算机之间直接通信，如计算机和PLC之间的通信，就不需要任何中间外围设备。因此，连接必须用串行零调制解调器型的电缆，因为无论是PLC还是计算机，都需要用2号引脚传送数据，用3号引脚接受数据。374～375

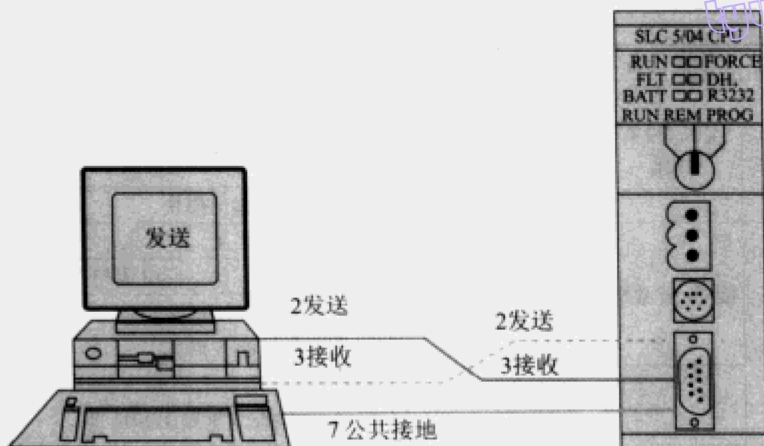


图 13-18 个人计算机和 PLC 处理器之间，采用零调制解调电缆进行的串行通信连接

思考题

1. 为什么 PLC 一般都要放在防护外罩内?
2. 用什么方法可以使防护外罩内的温度保持在允许的范围内?
3. 陈述两种电噪声耦合到 PLC 控制系统的方式。
4. 列出 4 种可能产生噪声的设备。
5. 描述出两种可以帮助降低电噪声的布线方式。
6. a. 哪些类型的现场输入设备和输出模块，在它们处于关断状态时，最有可能出现漏电流?为什么?
   b. 怎样减少漏电流?
7. 当 PLC 的电源线电压波动很大时，如何解决?
8. a. 在什么情况下，感性负载会产生一个大的电压峰值脉冲信号?
   b. 使用什么抑制直流负载?
   c. 使用什么抑制交流负载?
9. a. PLC 程序编辑功能的作用是什么?
   b. 查找功能作为编辑过程的一部分，作用是什么?
10. a. 解释离线和在线编程的不同之处。 b. 哪一种方式更安全? 为什么?
11. 列写出 4 种 PLC 的预防性维护工作。
12. a. 陈述两种要对 PLC 安装进行正确接地的重要理由。
    b. 列出 4 种对 PLC 系统安装时重要的接地措施。
13. a. 列出 4 种 PLC 处理器自检的内容。
    b. 当 PLC 处理器配有故障继电器时，电路通常是如何工作的?
    c. 看门狗电路的主要功能是什么?

376

14. 接地环路干扰是如何引起的?
15. 解释金属氧化物可变电阻缓冲器的工作原理。
16. 列出 4 种数据监控的常用功能。
17. 触点直方图功能提供了什么信息?
18. 在对设备进行维护或修理期间，大部分公司是如何确保不让设备运行的?
19. 尽管梯形图程序不容易出错，但它可能出现的故障类型是什么?
20. 解释在故障诊断过程中，怎样使用强制功能?

21. 当处理器扫描遇到暂停指令，会出现什么情况？
22. 解释中止指令的功能。
23. 错误的接线和连接以怎样的方式影响传送到 I/O 模块和由它们传出的信号？
24. 怎样判断现场设备和 I/O 模块是否兼容？
25. 概括 PLC 系统试运行要遵循的步骤。

习题

1. 如果 PLC 防护外罩的门没有关闭，会引起哪些问题？
2. 通常输出模块中会装有一个熔断器，说出这样做的两条原因。
3. 当 PLC 系统旁边的起重机从静止开始启动，PLC 系统会暂时工作不正常，引起故障的原因是什么？
4. 在检测系统时，用编程设备把一输出强制通电，但不是期望的输出指示灯变亮，出现这种情况的原因是什么？
5. 连接到模块的输入设备动作，但是对应状态指示灯没有亮，检查输入电压，没有电压存在，指明出现这种情况的两种原因。

377

6. 输出被强制通电，模块上面状态指示灯亮，但是输出设备没有运行，检查输出电压，电压正常，指明出现这种情况的两种原因。
7. 输出被强制通电，但是模块上面的状态指示灯没有变亮，检查出输出电压远低于正常水平，应该先检查什么？
8. 连接到 PLC 高频输入端的电传感器，产生假的输入信号，此故障怎样解决？
9. 输入状态指示等亮，但是编程设备监视器上面，处理器又没有接受到输入信号，如果更换了模块问题还没有解决，可能存在问题的其他两个地方是哪里？
10. a. 在一机器运转周期中，用一无磁限位开关来检测一状态的 5 次通断，如何通过观察 LED 状态指示灯判断限位开关是否正常工作？
    b. 怎样通过观察编程设备监控器，判断限位开关工作正常？
    c. 怎样通过观察 LED 状态指示灯，判断限位开关是否保持开状态？
    d. 怎样通过观察编程设备监控器，判断限位开关是否保持开状态？
    e. 怎样通过观察 LED 状态指示灯，判断限位开关是否保持闭合状态？
    f. 怎样通过观察编程设备监控器，判断限位开关是否保持闭合状态？
11. 假设在投入 PLC 系统运行之前，需要确认每一个输入设备都正确的连接到输入端子，输入模块或输入点都正常。说出进行这些测试的最安全的方法？
12. 假设在投入 PLC 系统运行之前，需要确认每一个输出设备都正确的连接到输出端子，输出模块或输出点都正常。说出进行这些测试的最安全的方法？
13. 如图 13-19 所示的梯形图程序，通过增加指令对程序进行更改，使得当泵 1 运行期间，泵 2 不能运行，并且当这一状态发生时，程序中止运行，把中止 ID 号 6549 送到 S2:7。

378

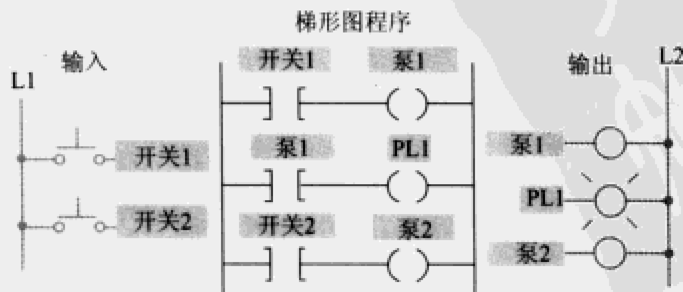


图　13-19

14. 图 13-20 所示的程序，执行该程序，当输入 A 闭合时，先断电 PL1 5s，然后通电 PL1 10s。

a. 检查梯形图程序，描述电路是怎样按照程序工作的。

b. 检查程序，对它进行修改使其工作正常。



图 13-20

379

第 14 章　过程控制和数据采集系统

学习目标：

- ❑ 讨论连续过程、批处理生产和个体产品生产的运行机制。
- ❑ 比较单机、集中和分布式控制系统。
- ❑ 解释过程控制系统主要部件的功能。
- ❑ 描述控制系统中开关、比例、积分和微分的作用。
- ❑ 概述出数据采集系统不同部分的功能。
- ❑ 讨论与数据采集系统选择、运行和连接相关的常用术语。

本章介绍了 PLC 控制的工业过程处理。数据采集作为控制过程的一个环节也包括在内。本章还描述了开环控制系统和闭环控制系统，并讨论了各自的功能特性。 380

14.1　过程处理的类型

过程控制涉及对控制系统的自动调整。根据被控过程的复杂程度，过程控制可以采取多种方法。PLC 具有数学运算和处理模拟量的功能，因此它非常适合于过程控制。过程控制系统的典型应用领域有：车辆装配、石油化工生产、石油提炼、电力生产和食品加工等。任何一种需要对一个或多个变量进行处理或控制的操作，都属于过程控制系统。通常过程控制中的变量包括：温度、速度、位置、流速、压力和液位等。现代制造工业的过程控制可以分成 3 种基本类型（即操作类型）：

- ❑ 连续过程
- ❑ 批处理生产
- ❑ 个体或离散产品生产

连续过程，就是把原材料送入系统的一端，在系统的另一端产生成品。处理过程连续运行，一旦处理开始，就要持续相对比较长的时间。根据过程的不同，时间可以用分钟、天，甚至是月来计量。在很多情况下，各部件都是顺序安装，呈流水线的样式通过各个站。各部件通过像传送带这样的机械传送设备，从一个站移动到另一个站进行装配。有些特殊的安装，仅用到手动操作，或者仅包括机器操作。

图 14-1 所示为引擎装配线的连续过程。引擎块被送入控制系统的一端，成品引擎从系统的另一端出来。当产品材料通过生产流水线时，会受到不同的处理（这个例子中，有装配、调整和检查环节）。自动装配要使用自动化机器或机器人给每一个站提供所需要的部件。 381 ~ 382

在批处理过程中，没有产品材料从过程的一个段流动到另一个段。材料被分批地加入过程当中进行处理，产生成品或者是还需进一步加工的中间产品。操作完成后，完成的产品被存储，而另一批产品被产生，每一批的产品可能不相同。许多化学相关的产品的生产都采用批处理的方法。

烘烤蛋糕所采取的步骤，就是一个很好的批处理例子。烹饪步骤包括：添加配料、搅拌配料、浇灌到烘盘，最后在特定的温度下烘烤一定的时间。工业的批处理过程跟这个相似，但是规模很大。采用批处理生产的产品有：食品、饮料、药品生产、油漆和化肥。图 14-2 所示为一批处理系统，两种配料加在一起、混合、加热，然后加入第三种配料，处理完以后存储起来。每次处理可能都有不同的特性。

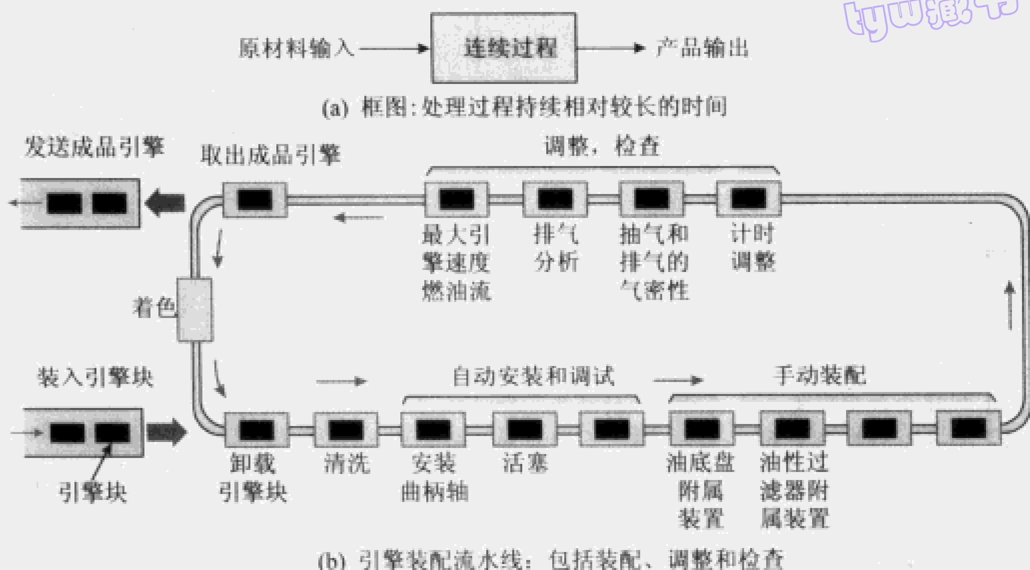


图 14-1 连续过程

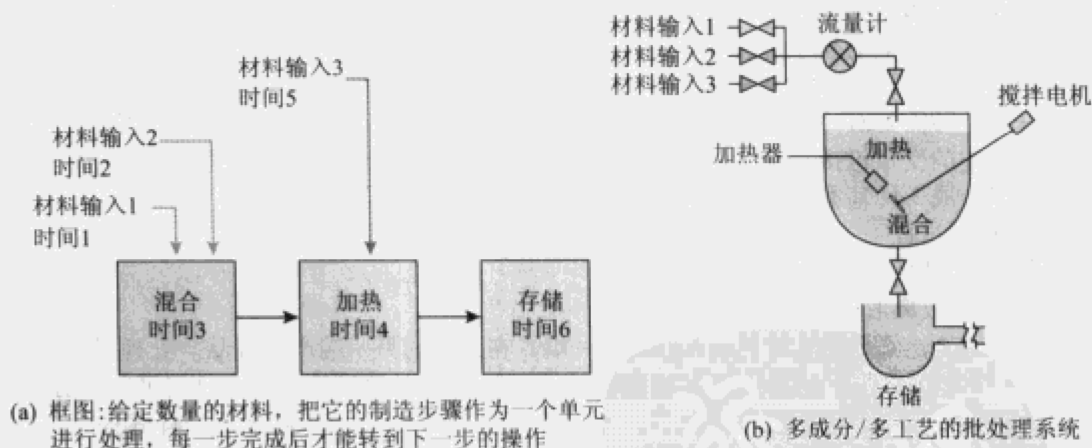


图 14-2 批处理

个体或离散产品生产处理在所有的处理系统中最为普遍。在制造生产过程中，通过一系列的处理，生产出有用的产品。过程处理包含以下几步：成形、钻孔、焊接等。工作部件通常都是离散的部分，必须放在单独的底座上加工。图 14-3 所示为个体产品生产处理系统。

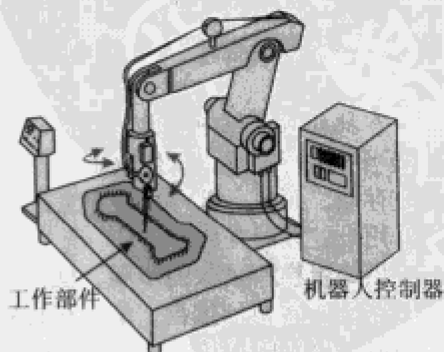


图 14-3 个体产品生产

在现代自动化的工厂，操作员只需设定控制系统并启动，而机器的运行都是自动完成的。这些自动化的机器用于大批量生产产品、控制非常复杂的操作或控制机器长时间、高精度的运行。它们代替了很多人类的决策、干预和监视。

机器最初都是通过机械控制的，后来发展到电动机械化控制，现在通常是纯电气化控制。电气控制就是通过 PLC 或计算机进行控制。对机器或过程

的控制可以分为以下几类：

- 电动机械控制
- 硬接线电气控制
- 可编程的硬接线电气控制
- PLC 控制
- 计算机控制

控制结构包括：单机控制、集中控制和分布式控制。单机控制用于控制一个单独的机器。这种类型的控制一般不需要和其他的控制器通信。图 14-4 所示为一单机控制的应用，这套系统适合于生产户内和户外使用的铝栏杆。操作员通过操作控制面板，输入给进的长度和数量，然后按下启动按钮启动操作。横杆的长度不尽相同，操作员需要选择横杆的长度和数量进行切割。

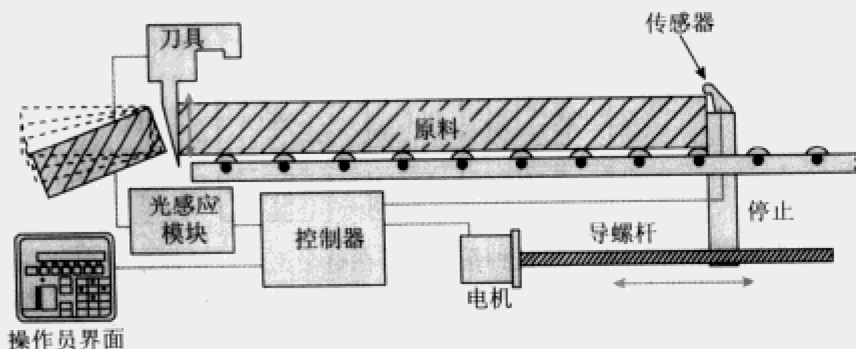


图 14-4　单机控制

集中控制，就是用一台中央控制器控制几台设备或几个进程（如图 14-5 所示）。控制结构采用一个单独的大型控制系统，控制许多制造环节或操作的驱动器。控制过程中的每一步都要通过中央控制器控制。控制器之间没有状态和信息的交换。由于有些过程被分成小的单元进行控制，复杂性会很高，因此需要进行集中控制。集中控制的一个缺点就是一旦中央控制器出现故障，整个过程就要停止。对于过程之间相互依赖并且许多不同的过程都要求高效利用设备和原材料的一些大型工厂来说，采用集中控制的优势就很明显。集中控制可以集中监控报警和改变操作，因此就可以免除在车间之间穿梭。



图 14-5　集中控制可以看作应用于大型控制过程或多个机器的大型单机控制

384

分布式控制系统（DCS）与集中控制系统不同，每台机器都是通过专有的控制系统控制。每个专有控制系统都是完全独立的，如果不需要它担当制造功能时，可以直接从控制框架中去除。分散控制由两个或多个计算机之间的通信来完成所有的控制任务。控制类型一般采用局域网（LANs），多个计算机分别就地控制不同的生产阶段或过程，定期交换信息和传送过程的状态信息。计算机之间通过单独的同轴电缆或是光纤以很高的传输速率进行通信。分布式控制把控制器和 I/O 模块安放在距离被控制机器很近的地方，极大地减少了现场接线，并增强了控制性能。

分布式控制系统灵活性好，对于需要大量的批处理和连续作业的系统来说是很好的选择。分布式控制系统，允许在多个控制单元之间进行处理任务的分配。每个回路控制器被

放在靠近被控制点的地方，并且具有处理能力；而不是仅由一台位于中心控制点的计算机来处理所有任务。

图 14-6 为一台主机监控的 DCS 系统。主机可以是一台个人计算机，能处理如上传和下载程序、报警、数据存储等任务，还可以作为操作界面。用来监控系统的远程操作计算机，一般被安装在远离工业环境、灰尘少并且温度可以控制的地方。每台 PLC 控制相关的机器或过程。根据不同的控制过程，大多数情况下，一台 PLC 发生故障不会停止整个控制系统。

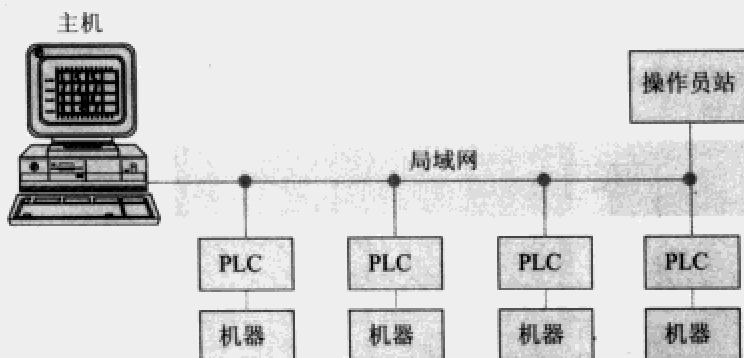


图 14-6 分布式控制系统（DCS）。这种系统被用在整个工厂或是控制过程被分为几个子系统的时候，每个子系统都由单独的 PLC 控制，PLC 之间又相互连接构成一个独立的控制实体。控制器之间的通信连线有多种形式

385

14.2 控制系统结构

过程控制系统，可以被定义为改变材料物理特性或化学特性所需的功能和操作。过程控制，一般指的是在工业领域中对产品的制造或生产。利用 PLC，通过用户程序，对机器、过程进行操作和监控。图 14-7 所示为过程控制系统的主要组成部分，包括下面几个部分。

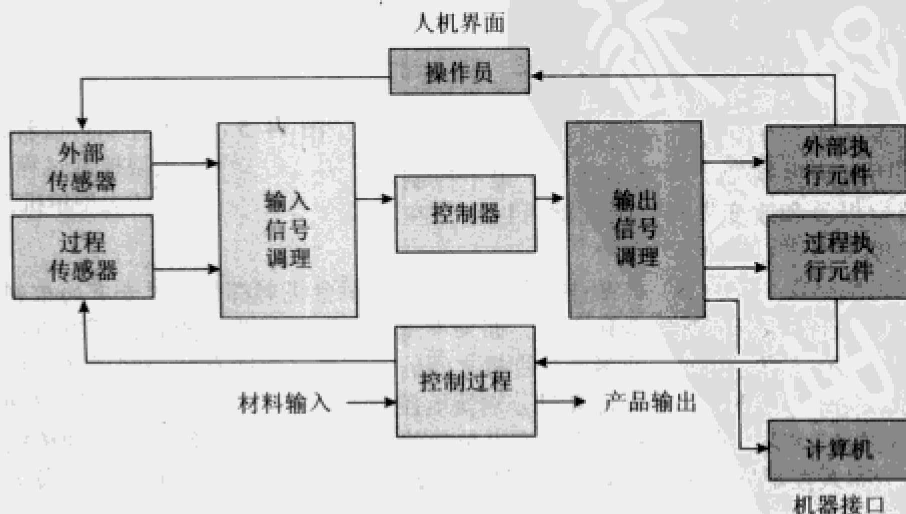


图 14-7 过程控制系统的组成

1. 传感器

- 从过程内部或外部环境提供输入信号。
- 把如压力、温度、流速和位置这样的物理信号转换为电信号。
- 反映物理信号的变化，因此产生的电信号可以用来监视和控制过程。

2. 人机界面

- 允许人为设置初始状态或改变过程操作。
- 允许人为通过各种开关、控制装置和键盘输入信号。
- 根据提供的输入信号操作，输入信号包括：急停、改变速度、运行模式、块的数量和分批混合的方法。

3. 信号调理

- 涉及把输入和输出信号转换为可用的形式。
- 包括对信号进行调理，如放大、衰减、过滤、定标、A/D 和 D/A 转换。

386

4. 执行元件

- 将系统输出电信号转换为物理动作。
- 过程执行元件包括：流量控制阀、泵、位置驱动器、变速驱动器、离合器、制动器、螺线管、步进电机和功率继电器。
- 通过外部执行元件，如仪表、CRT 监视器、打印机、报警器和指示灯，监视过程变量的状态。
- 能够把控制器输出信号直接传送到上位机，进行数据存储和结果分析（机器接口）。

5. 控制器

- 根据系统输入信号做出决策。
- 产生输出信号，控制执行元件，实施决策。

控制系统可分为开环控制系统和闭环控制系统。图 14-8 为一典型的开环控制系统。该控制系统通过输入期望设定值到控制器（也就是命令、参考值），使系统达到理想控制效果，并且接收输出结果。因为输入到控制器的只有设定值，很明显开环控制系统对过程进行控制存在盲目性，也就是说，控制器没有接收到任何有关过程当前状态的信息，也没有做出任何调整。开环控制系统与闭环控制系统相比，降低了复杂程度和成本。但是，开环控制系统存在不确定的因素，而且控制精度也没有闭环系统高。



图 14-8　开环控制系统

开环控制系统，在许多工业应用场合仍然可以看到。图 14-9 所示为 PLC 控制的步进电机开环控制系统实例。步进电机通常在小功率、低速度的条件下，控制位置的移动。步进电机就是有几组线圈绕在转子上面的永磁电机，线圈被称为相（A 和 B）。线圈接在 PLC 的输出端，在用户程序的控制下轮流被激励。转子的转动由哪一相被通电决定。PLC 没有接收转子的反馈信号，就不能监视转子是否转动，但我们还是认为电机能够正确的旋转，因为转子线圈被步进脉冲顺序驱动。开环控制或无反馈控制同系统的下载和独立组件一样平稳。

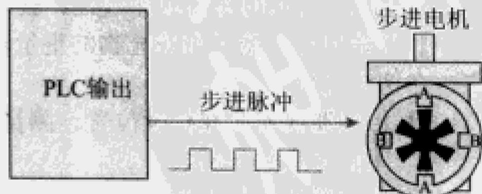


图 14-9　开环步进电机控制

闭环控制系统的过程输出反过来影响输入控制信号。系统测量实际的过程输出并与期望的输出相比较。控制系统连续调整，直到期望输出与实际输出的差值在实际足够小的范围内。虽然闭环控制系统比开环控制系统更难调整或校正，但是它能更精确地确定过程输出的发生，而且更精确地监视过程的输出。图14-10所示为典型的闭环控制系统。通过测出实际的输出，并反馈回来与输入设定值相减，决定系统输出（因此也称为反馈控制）。如果存在偏差，控制器就不断调整输出，直到偏差为0。系统各个部分如下。

387

1. 给定值

- 输入信号，决定了过程的期望输出。
- 通常由操作员提供，也可以由一些电路提供。

2. 过程变量

- 包含当前过程状态信息的信号。
- 一般是反馈信号。
- 理想值与给定值相等（说明系统在期望的情况下工作）。

3. 偏差放大器

- 决定过程输出是否与设定值相等。
- 通常是差动放大器提供输出信号，被称为*误差信号*或*系统偏差信号*。
- 偏差信号的大小和极性，决定了过程控制的方式。

4. 控制器

- 根据输入偏差信号，产生合适的校正输出信号。

5. 输出执行元件

- 直接引起过程改变的元件。
- 例如电机、加热器、风扇和螺线管等。

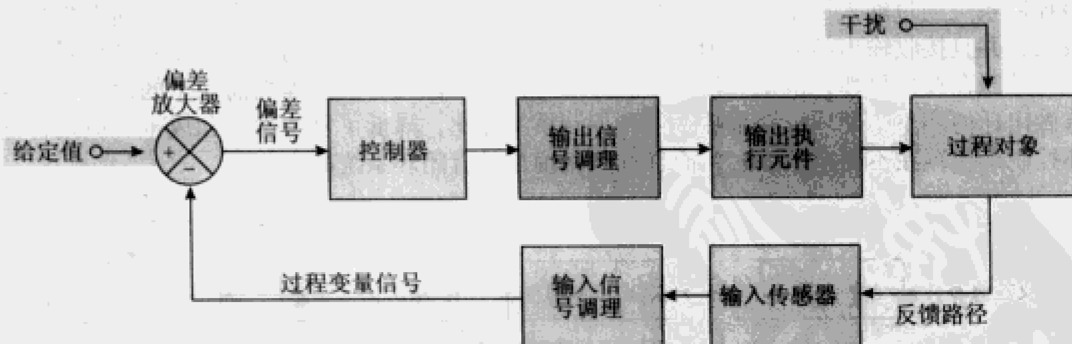


图14-10 闭环控制系统

图14-11所示的容器填充过程，就是一个连续控制过程的实例。一个空箱子被移动到位后，开始填充。箱子和填充物的重量被监控，当实际的重量达到设定值的时候，填充结束。在这个过程中：

388

- 传感器被安装在标度位置，测量容器的重量，产生反映容器与填充物总重量的电压信号或数字码。
- 传感器信号与事先设置好的反映期望重量的电压或数字码信号相减。
- 只要输入信号和反馈信号的偏差大于0，控制器就会一直保持电磁阀门打开。
- 当偏差变为0时，控制器输出一个信号，关闭电磁阀门。

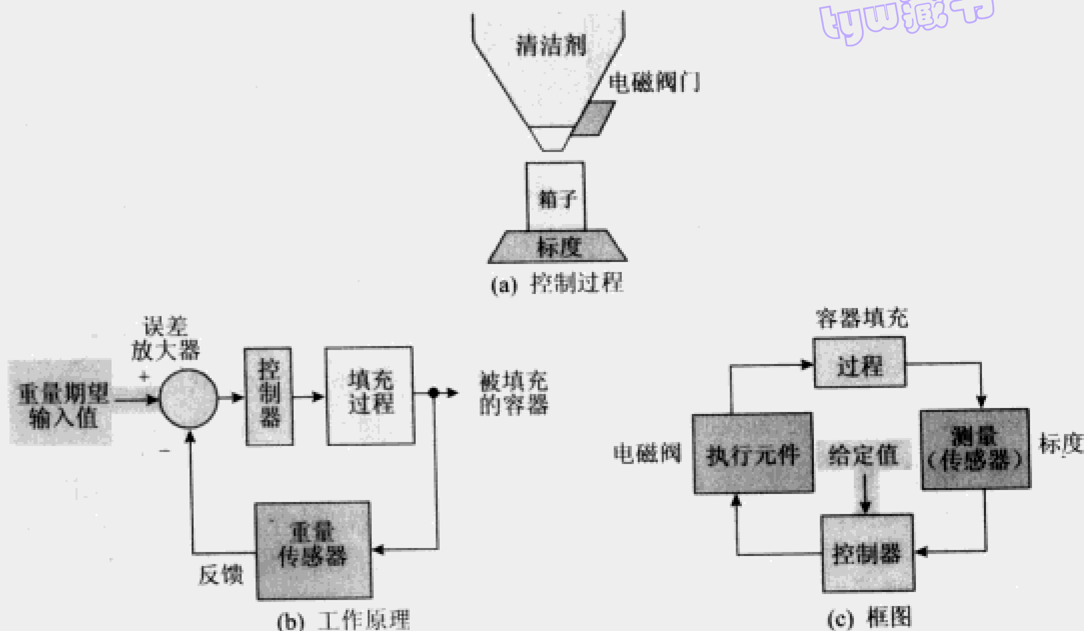


图 14-11 容器填充闭环控制过程

14.3 控制器

控制器有几种分类方法。举例来说，根据它们动力源的类型可分为两种类型，一种是电动控制器，一种是气动控制器。气动控制器是通过空气压力动作的执行装置，电动（电子）控制器通过电信号进行操作。

控制器也可以根据它们的控制类型进行划分。在本章中，我们讨论根据这种方式划分的 4 种类型：

- 通/断
- 比例（P）
- 积分（I）
- 微分（D）

通/断控制（也被称为双位置控制和继电器控制），末级控制元件不是接通就是断开——一种是变量测量值大于设定值的情况，另一种是变量测量值小于设定值的情况。末级控制元件不可能处在一个中间位置，通过通－断循环动作实现控制功能。

图 14-12 所示为利用通/断控制的液体加热系统。如果液体温度低于设定值，蒸汽阀门打开，蒸汽流入；当液体的温度低于设定值时，蒸汽阀门关闭，蒸汽被隔断。只要系统在运行，就一直循环执行通断动作。

图 14-13 所示为通/断温度控制的控制响应曲线。当温度低于设定值时，输出接通；当温度达到设定值时，输出被关断。控制虽然简单，但是在设定值出现超调和反复的时候，对于有些系统是很不利的。输出变量在设定值附近以一定幅值和频率振荡，幅值和频率大小取决于系统的控制能力和响应时间。这种振荡是通/断控制器的典型特征。通过增加控制器的灵敏度，可以降低振荡的幅值，但是这会引起控制器接通和关断的频率变高，也是通常不希望出现的结果。

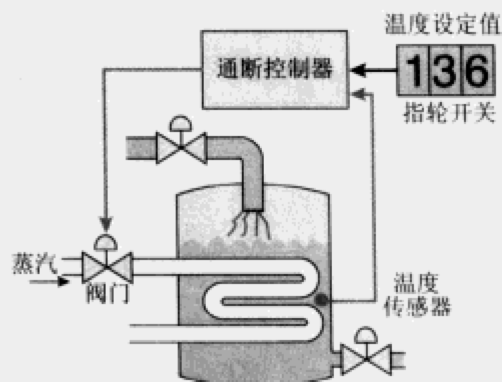


图 14-12　液体加热通/断控制系统

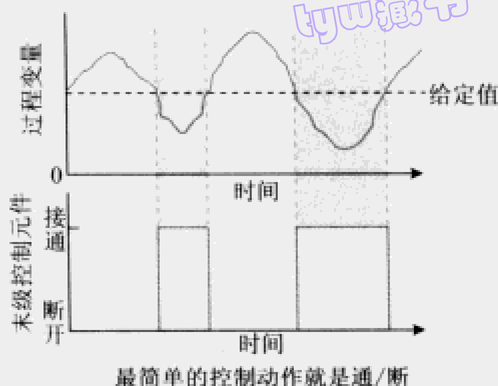


图 14-13　通/断控制

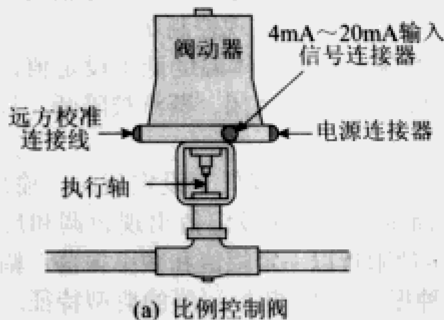
通常在给定值附近设置一死区。控制器的死区一般是可选择的值，决定了在给定值以上或以下的偏差范围，只要过程变量值在设置范围内，控制器不产生输出。死区避免了控制设备在给定值附近产生不规则的振荡。当小的波动引起控制器对位置进行小的调整时，控制器就会产生振荡。通/断温度控制通常用在下列场合：

390

- 控制精度要求不高时
- 设备通/断频率不高的系统
- 温度改变缓慢的大滞后系统
- 报警系统

比例控制可以消除通/断控制的反复和不规则振荡，允许末级控制元件处于接通和关断之间的中间位置。比例作用可以根据测量变量值与期望值偏移量的大小改变控制过程执行元件模拟量控制值。

比例控制可以根据误差信号的大小，更加准确的控制过程变量的变化，因为它的输出可以是全通和全断之间的任何一个值。图 14-14 所示为作为执行元件的电机驱动模拟量比例控制阀。通常地，执行元件从控制器接收 4mA～20mA 的输入电流信号，并驱动阀门做线性移动。4mA～20mA 信号回路是在工业过程控制环境下，远距离传输模拟量信号常采用的方式。4mA 对应于最小值（通常是 0mA），20mA 对应于最大值（满量程）。4mA 的最低限，可以使系统检测阀门是否打开，如果电路开启，会产生 4mA 的电流，为系统显示报警用。因为信号是电流型的，它与电压信号相比，不容易受连接线路阻抗变化和来自其他信号的噪声干扰的影响。通常使用双绞线来降低来自电动机、变压器等磁场干扰的影响。



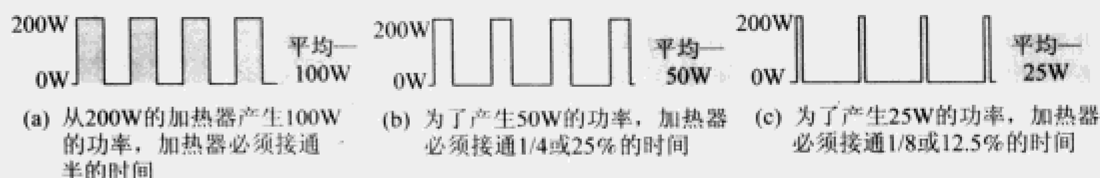
(a) 比例控制阀

执行元件电流 (mA)	阀门响应 (%打开)
4	0
6	12.5
8	25
10	37.5
12	50
14	62.5
16	75
18	87.5
20	100

(b) 阀门的响应与对应的执行元件电流

图 14-14　电机驱动的模拟量比例控制阀

比例作用，也可以通过短暂间歇性地接通和关断末级控制元件来实现。这种时间比例（也被称为比例脉宽调制）控制，改变接通时间与关断时间的比率。图 14-15 所示为时间比例控制的从 220W 加热器获得变化功率的例子。 391



(a) 从200W的加热器产生100W的功率，加热器必须接通半的时间 (b) 为了产生50W的功率，加热器必须接通1/4或25%的时间 (c) 为了产生25W的功率，加热器必须接通1/8或12.5%的时间

图 14-15 200W 加热器的时间比例控制

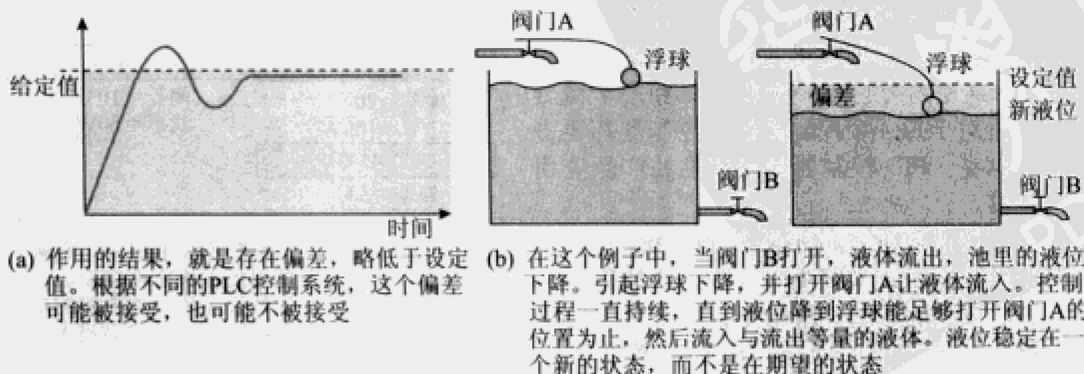
如图 14-16 所示，比例部分在给定值附近的比例区域内起作用。在这区域之外，控制器只实现全通或全闭的功能，输出只能是完全打开（在区域以下）或完全关闭（在区域以上）。可是在区域以内，输出接通和关断的比率，则根据测量值与设定值的偏差决定。在给定值（比例区域的中心点），输出接通和关断时间比率为 1:1，即接通时间和关断时间是相等的。如果温度值与设定值存在偏差，输出接通和关断时间与温度偏差成比例。如果温度低于设定值，输出保持常通，如果温度太高，输出保持常闭。

时间比例			温度（°F）	4mA~20mA 比例控制	
百分比	接通时间(s)	关断时间(s)		输出电流	输出百分比
0.0	0.0	20.0	超过540	4mA	0.0
0.0	0.0	20.0	540.0	4mA	0.0
12.5	2.5	17.5	530.0	6mA	12.5
25.0	5.0	15.0	520.0	8mA	25.0
37.5	7.5	12.5	510.0	10mA	37.5
50.0	10.0	10.0	500.0	12mA	50.0
62.5	12.5	7.5	490.0	14mA	62.5
75.0	15.0	5.0	480.0	16mA	75.0
87.5	17.5	2.5	470.0	18mA	87.5
100.0	20.0	0.0	460.0	20mA	100.0
100.0	20.0	0.0	低于460	20mA	100.0

例子：加热系统。设定值：500°F。比例区域：80°F（±40°F）

图 14-16 比例区域

理论上讲，比例控制能满足过程控制的所有需要。系统输出产生任何变化，控制器的输出都会做出适当的变化对其进行调整。然而，比例控制器会导致过程偏差，被称为偏移量或固定偏差。系统稳态误差是控制器实际值与期望值之间的偏差，如图 14-17 所示。当系统过程状态发生大的改变时，可能需要操作员进行小的调整（手动），才能使控制过程变量设定在初始值。比例控制通常与微分或积分控制结合起来使用。 392



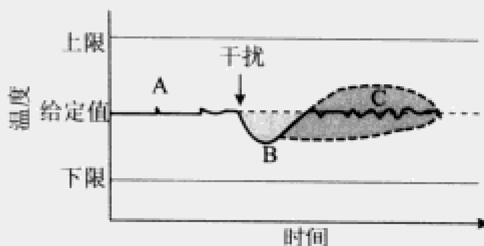
(a) 作用的结果，就是存在偏差，略低于设定值。根据不同的PLC控制系统，这个偏差可能被接受，也可能不被接受 (b) 在这个例子中，当阀门B打开，液体流出，池里的液位下降。引起浮球下降，并打开阀门A让液体流入。控制过程一直持续，直到液位降到浮球能足够打开阀门A的位置为止，然后流入与流出等量的液体。液位稳定在一个新的状态，而不是在期望的状态

图 14-17 比例控制稳态误差

积分作用，又称为无静差调节，反映偏差信号的大小和持续时间，因此积分控制器的输出是偏差的数学积分。当过程变量值与给定值不相等时，偏差信号就产生了，因此积分作用能引起输出改变，并持续到偏差消失。积分作用能消除稳态误差。积分作用可以用每重复一次所用的分钟数或每分钟的改变量来计量，反映的是改变量与时间的关系。

微分模式控制器，反映的是误差变化率的大小，也就是当误差改变越大，调节输出也越大。微分作用通过时间常数测量。

比例积分（PI）控制（如图14-18所示）结合了这两种控制类型的特征。测量值的每一步改变，引起控制器的输出成比例的变化，紧接着积分响应又加到比例响应中。因为积分作用使控制器输出的改变作为时间的一个函数，所以积分作用越强，输出的改变就越快。PI控制模式用于过程变量改变不频繁，或是有改变时改变量也很小的系统。



- 为了消除偏移误差，控制器需要改变它的输出，直到过程变量偏差为0
- 无静差（积分）控制，根据过程变量与给定值之间的差值，通过改变控制器的输出进行调节
- 积分作用以后，新的平衡点在C
- 因为比例控制器必须在比例区域内运行，因此比例区域应该移动到包含C点的范围内，带有积分作用的控制器能自动实现这个功能

图14-18 比例积分（PI）控制

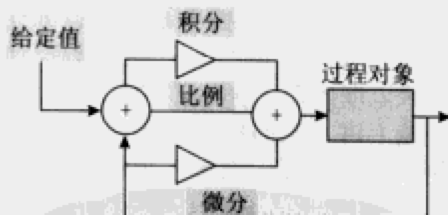
与积分类似，微分也作用于偏差信号，只不过微分是偏差改变速率的函数，而积分是误差的累计值。微分在一定可选择的时间间隔内输出变化量，时间间隔通常用分钟表示。由微分引起的控制器输出端的改变量，是通过输入改变量的微分计算出来的。用输入变化量而不用比例控制器误差的改变量，可以增强系统的响应性能。微分能够快速定位输出，而单独的比例控制器只有到最后才能定位输出。实际上，微分是通过快速转移到比例区域内，抑制偏移或误差的产生而起作用的。

比例微分（PD）控制，用于偏差变化速度快的过程控制系统中。把微分控制加到比例控制中，控制器输出就能响应变化量的速度及其大小。

比例－积分－微分（PID）控制器，根据系统误差信号的大小、持续时间和变化速率，产生输出（如图14-19所示）。系统突加的干扰，欲破坏系统原来的状态。PID控制器比其他任何控制器，都能更迅速地将系统的偏差减少为0，这是因为它有一个积分器和一个微分器，但是对于每一个控制系统，控制器都要进行常规调整。PID控制器是最成熟并广泛应用的控制器。

为了快速和准确的消除控制系统的波动，通常采用数学公式来解决。这些公式被称为算法。算法就是固定的一组指令，用来在一定的步内解决特定类型的问题。在PLC控制系统中，CPU执行这些指令，处理被发送到过程执行元件的数据。

393



- 在设置期间，设定好给定值、比例区域、无静差（积分）值、比率（微分）和输出限幅
- 所有这些参数都可以在系统调整期间进行修改
- 积分能提高控制精确性，微分能降低瞬时性干扰的超调
- 输出可以用来控制阀门位置、温度计、流量计等设备

(a) PID控制回路

T=30℃	T=40℃	T=45℃
偏差=20℃	偏差=10℃	偏差=5℃
功率=100W	功率=50W	功率=25W

(b) PID允许输出"功率"等级不相同。举例来说，假设窑炉的温度被设定在50℃。一旦温度值低于50℃的设定值，加热器的功率就要增加。温度越低，加热功率就越大。当温度信号接近温度值时，PID能平缓的调整输出功率

图14-19 PID控制

任何长期运行的系统，无论是大系统还是小系统，在输入和输出之间都要保持质能平衡。如果系统一直都在平衡状态下运行，控制就很简单。但是系统经常发生变化，过程控制的重要参数是时间，即任何输入的变化引起输出改变所需要的时间。系统时间常数，可以从几秒到几小时不等。PID 控制器能根据不同的过程时间常数做出调整，因此能及时处理过程的变化。PID 控制器根据偏差值和信号变化速率值，以特定的数学方式，调整控制器的输出。

实现 PID 控制的常用 PID 公式为：

$$Co = \overbrace{K \quad (E}^{\text{比例}} \quad + \quad \overbrace{1/Ti \int_0^t E\,dt}^{\text{积分}}$$

$$+ \overbrace{KD \quad [E - E(n-1)]/dt\,)}^{\text{微分}} \quad + \text{bias}$$

这里，Co——控制输出

K——控制增益（没有单位）

$1/Ti$——积分增益常数（每分钟的改变量）

KD——微分增益常数（分钟）

dt——采样时间（分钟）

bias——输出偏置值

E——偏差，等于给定值减去测量值

$E(n-1)$——上一次采样的偏差

PLC 可以安装带有 PID 控制功能的输入/输出模块，也可以用自带的数学函数实现 PID 控制功能运算。PID 本质上就是一个公式，控制器利用它处理数据变量。图 14-20 所示为 PCL 在 PID 控制回路的使用情况。检测控制变量（压力）并产生反馈。PLC 的用户程序，通过比较反馈值与设定值，产生偏差信号。偏差信号通过三种方式检查：比例、积分和微分。然后，控制器产生一个命令（控制输出），通过改变出水阀门位置，调整偏差。 394

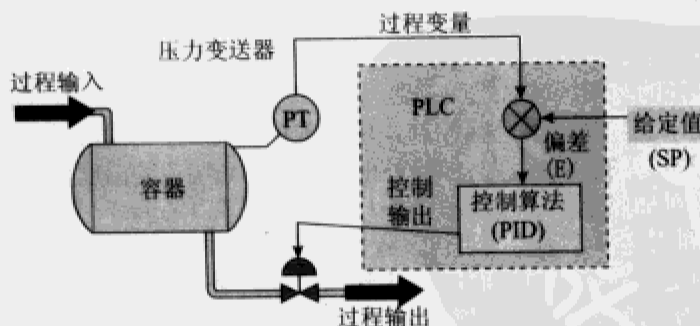


图 14-20　PLC 用于 PID 控制回路

PID 控制回路的响应，就是通过调整输出来补偿偏差的速率。PID 控制回路通过改变比例增益、积分增益和/或微分增益实现调节或调整。PID 回路的测试，通常使给定值突变，然后观察系统的响应速度。可以通过以下方式进行调整：

- 比例增益增加时，控制器响应速度变快。
- 当比例增益过高，控制器会变得不稳定，产生振荡。
- 积分增益起稳定器的作用。
- 即使偏差为 0，积分仍然起作用（例如，当炉子的温度达到设定值，还是需要功率来维持温度恒定）。

- 如果没有这个基本功率，控制器的输出会低于设定值并产生振荡。
- 微分起超前校正器的作用。
- 当信号变化太快时，微分用来调整使其变慢。

基本上，PID 控制器的调整，包括调整增益（比例区域）、速率（微分）和复归时间（积分）参数（控制常数），使它们的值能满足控制系统的需要。根据过程变量值与给定值之间偏差的不同，对参数进行调整，改变控制器的输出，过程变量值也相应改变。一般来说，控制器的调整有三种方式：

1. 手动方式

- 操作员估计使系统达到预期响应的各参数的值。
- 根据特定的系统，用试错法对比例、积分和微分系数进行单独校正或调整。

2. 半自动化或自动调整

- 控制器进行计算并设置 PID 参数
  - 测量传感器
  - 计算偏差、误差的和与偏差变化速率
  - 根据 PID 算法计算期望输出
  - 更新控制器输出

395

3. 全自动化或智能化

- 这种方式在工业领域中也被称为*模糊逻辑控制*。
- 控制器利用人工智能方式，根据需要不断的调整 PID 的参数。
- 不同于利用公式计算输出，模糊逻辑控制器利用评价规则。第一步，把偏差和偏差改变量“模糊化”为语言变量，像“不大于”或“小于”等。利用简单的如果 – 那么规则评价产生输出，输出必须经过反模糊变为连续变量值，才能用来控制如阀门的位置等连续变化的物理量。

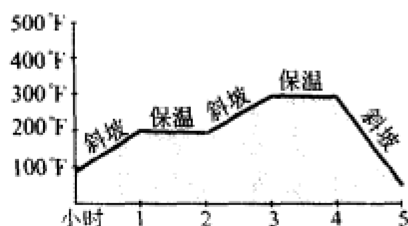


图 14-21　斜坡与保温图表

温度斜坡和保温控制图表如图 14-21 所示，通过选择一系列的设定值，单独对控制参数进行调整。设定值在端点之间的可选择的时间区域内线性变化。可以设置斜坡和保温段次序的重复周期数。

如果系统不要求对加热速率进行控制，用标准的给定值控制器就能实现温度的控制。只要产品和加热炉本身的容量允许，加热炉很快就能达到期望的温度值，但不一定是线性变化的。如果需要对加热速率进行控制（如：加热速率 1 °F/min），就需要用一个可编程的/斜坡控制器。利用这种控制器，可以设置特定的线性加热速率。

保温时间指的是产品在期望的时间内达到期望的过程温度值的时间。可以对 PLC 进行编程，使其在一定时间内保持期望温度不变，然后温度下降完成循环。对于大多数的应用，保温周期从加热循环完成的时刻开始。

PID PLC 输出指令，用于物理性能自动控制的闭环控制系统，如温度、压力、液位和流速等过程回路。图 14-22 所示为 Allen-Bradley SLC-5 PLC 的 PID 输出指令。这个指令从模拟量输入模块读取数据，通过算法对数据进行处理，然后提供输出信号给模拟量输出模块，从而有效的将过程变量控制在期望值。PID 指令能实现这些功能，它是 PLC 最复杂的指令之一。

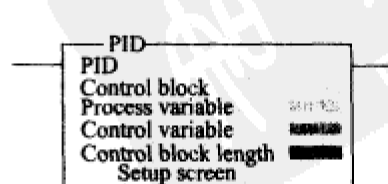


图　14-22

PID 利用模拟量输出、时间比例方法实现加热控制

或带有滑线反馈的阀门位置控制。设定值与过程变量值之间的偏差越大，输出信号就越大，反之亦然。可在控制输出加上一个附加值（前馈值、偏置）作为偏移量。PID 计算结果（控制变量）驱动过程变量，使其向给定值靠近。

396

一般地，PID 指令被放在没有逻辑条件的梯级上面。当梯级条件变为假时，输出还保持它的上一次值。当梯级条件变为假时，积分值被清零。这个指令的一些基本参数信息如下：

- **控制块**——存储执行指令所需数据的文件。
- **过程变量**——存储过程输入值的元件地址。
- **控制变量**——存储 PID 指令输出值的元件地址。
- **配置屏**——双击此指令，会出现一个显示框图，可以方便的在上面输入其他的参数值，实现 PID 指令的完全配置。

14.4 数据采集系统

数据采集（DAQ）基于计算机系统，实现对数据的收集、分析并存储。起初，这种系统只被用在装有成千个输入数据面板的大型计算机上面。但是今天，功能强大的个人计算机与 PLC 已经把数据采集系统的优势，扩展到所有的制造领域，并且造价不高。数据采集系统或计算机接口系统，允许从外部输入信号给计算机。产生信号的装置有：压力变送器、流量计等，它们把信号转换为处理器能够辨识的标准形式。通过数据采集系统，可以对数据进行收集、监视、显示和分析。监控和数据采集系统（SCADA）还有附加的控制外部输出的能力，可以更加高效地对系统进行精确控制（如图 14-23 所示）。

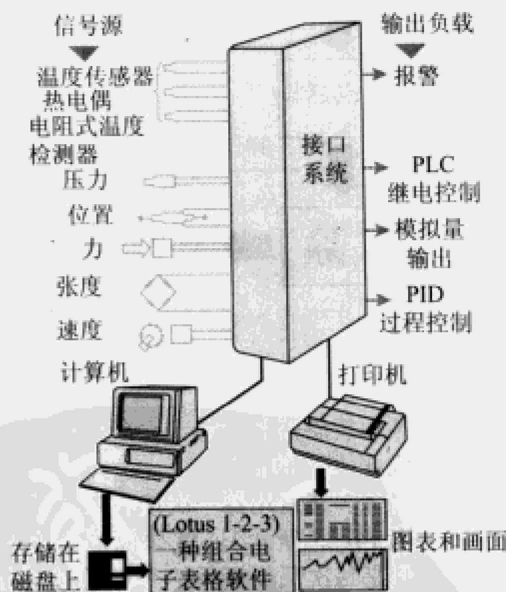


图 14-23 监控和数据采集系统（SCADA）

数据采集系统的最大优势，就是数据可以以一定的形式自动储存起来，以后需要分析的时候，可以正确地将其恢复，不需要做其他的工作。检查各个因素的作用，比刚开始预见的要容易，因为测量一些额外的值花费很低。在处理器的控制下，进行测量，然后将数据显示在屏幕上，并存储在磁盘中。这样很容易就能获得精确的测量值，并且测量速度没有机械方面的限制。

除了生产产品，PLC 也产生数据。储存在数据中的信息，经常被用来提高过程的控制效率。新一代的 PLC，可以接收宽范围的输入信号，包括模拟量和数字量，并且具有强大的算术运算功能和控制能力，因此现在 PLC 也适用于一些数据采集系统。在用户程序的控制下，PLC 提供字级的数字信号给数模转换器，把数字量转换为对应的模拟量。转换后的模拟量就可用于控制输出设备（如执行元件）或被记录下来。

397

数据采集软件与硬件接口设备一同工作。除了通过接口设备对数据进行采集外，软件还可以实现把数据显示在表格中，并对数据进行分析，像伸展表、直方图、圆形图、线性图等。大多数的数据采集系统都有基本的相似之处。

主要的系统元件包括以下几方面的内容（如图 14-24 所示）：

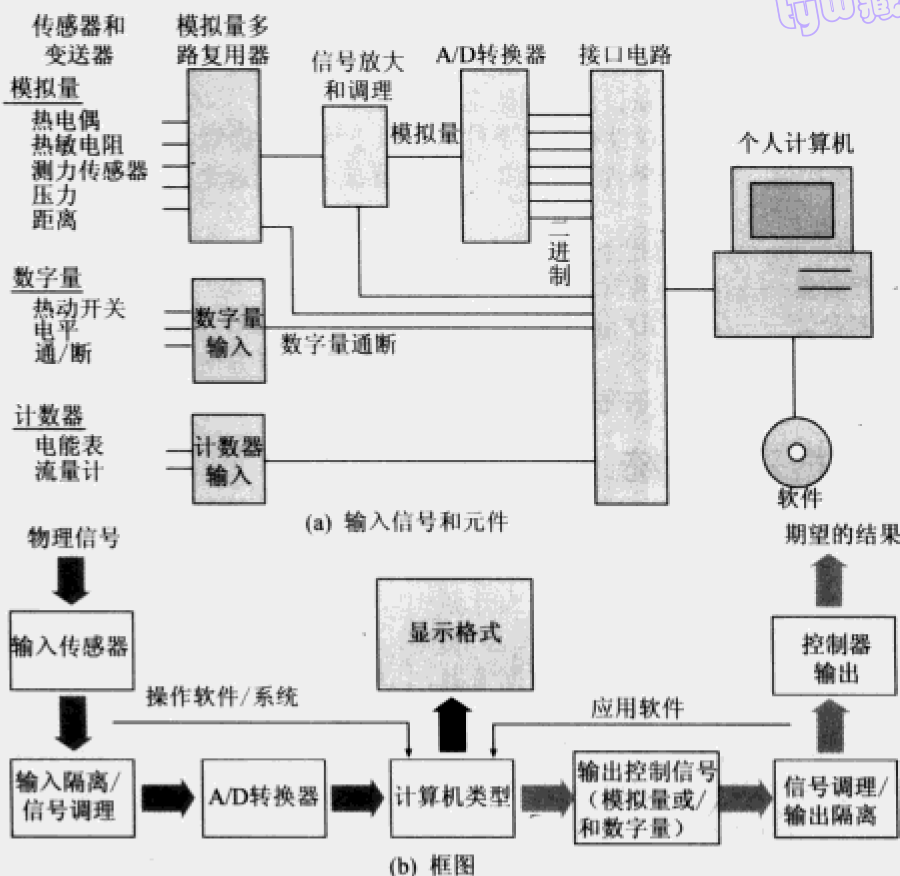
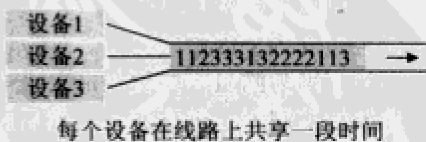


图 14-24　数据采集系统的主要元件

- 利用传感器和变送器，把机械信号或电信号转换为标准的信号，如：4mA ~ 20mA 或 0 ~ 5V 的信号。
- 时分多路复用器一次接收几路信号，但是只允许用户选择一路信号进行检查（如图 14-25 所示）。就如同电视有 99 个频道，但是每次只能显示一个。通过时分复用昂贵的设备，可以降低系统造价。多路复用系统，实际上就是利用智能转换器，把大量的并行线用简单的双绞线代替。这样既简化了接线，又降低了成本。

398



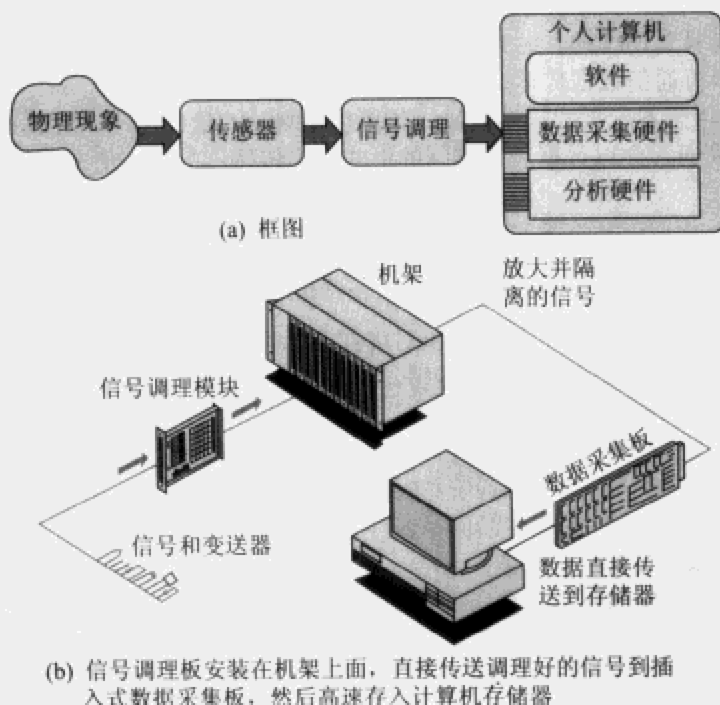
每个设备在线路上共享一段时间

图 14-25　时分多路复用器

- 放大器用来放大信号。放大器可以把传感器产生的电压或电流信号放大。
- 信号调理电路，使信号更加容易测量和更稳定（如电流电压转换器、V/F 转换器和滤波器等）。
- A/D 转换器是把模拟量信号转换为数字量信号的特定调节器。
- 数字量和计数器的输入信号，有一定数量的可能值。（1 bit 有 2 个可能值，4 bit 有 16 个可能值）。

数据采集接口系统有两种：*插入式*和*独立式*。这两种为数据采集系统提供信号调理的形式，或是在数据采集板上面直接放置信号调理电路，或是利用独立的已经嵌入了信号调理功能的信号采集设备。

图 14-26 所示为模块化的插入式数据采集系统，它利用独立的模块实现信号调理和数据采集。模块化系统的优势在于，它可以利用最新的插入式数据采集硬件、软件和计算机技术，而独立的数据采集接口系统的信号调理性能是保持不变的。 399



(a) 框图

(b) 信号调理板安装在机架上面，直接传送调理好的信号到插入式数据采集板，然后高速存入计算机存储器

图 14-26 模块化插入式数据采集系统

大多数的传感器在没有预处理之前，不能直接连接到数据采集板上。特殊的信号调理是必须的。举例来说，热电偶需要冷端补偿；电阻式温度检测器（RTDs）、热敏电阻和应变片需要电压或电流激励。大多数的传感器需要对信号进行放大，使其达到 A/D 转换器的输入范围。隔离和滤波能够保护设备，并能增加在工业环境下测量的精确性；抗重叠滤波器能有效滤除测量时的高频噪声。输入多路复用器，用于连接多路信号，经济实用。外部放大器把低电平信号放大，可以更好的向数据采集板传输。来自开关或大电流继电器的离散输入信号在送入计算机之前，也需要进行隔离和调理。

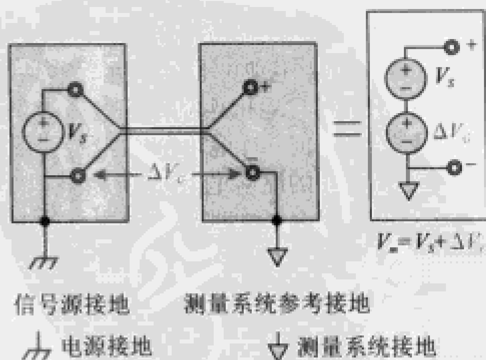


图 14-27 接地环路

数据采集系统能产生接地环路（如图 14-27 所示）。当在同一电气通路中出现两个或多个接地点时，在一些接地线之间会有不同程度的电流流过，这样就形成*接地环路*。数据采集板接收到的信号，有期望的信号电压值和额外的接地电流在系统中引起的电压值。如果其中一个参考接地点不去除，信号调理设备与隔离电路一起将隔断接地环路，这样在测量时就可以避免接地电流的影响。

与数据采集系统的选择、运行和连接相关的、重要的名词术语包括：

绝对信号 不需要参考量的信号。

精确性　不等于分辨率（通常认为这两者相同），精确性是误差的总和，由下面几个方面引起：

分辨率 + 增益 + 偏移 + 噪声

混淆　在采样数据中出现的、错误的低频信号，由采样速率太低引起。

模拟量　信号是连续的，并且有无穷多个可能值存在。

波特率　串行通信数据传输速率，用每秒多少比特表示（b/s）。

标定　确保传感器出来的输出信号，与传感器测量的量成比例（并且比例的大小是知道的）。制造厂家通常会以数据表或铭牌的形式给出传感器的标定值。例如：

位距传感器
满标度…………………………2.0mm
满标度时的输出…………5.0mV
激励电压值…………………10.0V

400

- 在这个例子中，输出5mV的电压对应在满标度2.0mm处。
- 偏移因数，常被叫做零位，是传感器设定在0mm处时的输出。

控制更新间隔　指的是计算机检查测量值并调整控制输出的频率。如果更新太慢，控制可能不起作用；如果比系统变化速度快太多，对计算机时间造成了浪费。更新速率一般设定为使控制系统期望精确性下降到所花时间的5倍到10倍。

数据存储值　数据采集系统可以保存各种各样的数据，包括：

- 平均值。所有数据的平均值。
- 最大值。在给定间隔内所测得的数据的最大值。
- 最小值。在给定间隔内所测得的数据的最小值。
- 原始采样。每个采样时刻的测量值。
- 总和。给定间隔内所测得的数据总和。

漂浮信号和共模电压　漂浮信号就是没有连接到其他任何系统或电压的信号。汽车标准的12V蓄电池电压没有任何连接线，是漂浮信号的一个实际例子：

- －端和＋端之间的电压为12V
- 每一端与其他任何系统之间的电压为0（或没有规定）。

比如一组蓄电池串联在一起，并且有一端接地（如图14-28所示）：

- 最开始与地相连的点的电压为0。
- 穿越任何一个蓄电池的电压差仅为5V。
- 电池组的终端电压为40V。
- 在第三个电池进行差动测量的共模电压值为10V。共模信号只有存在差动信号时才出现。共模电压值是在＋端与－端与参考接地点差动连接的电压的平均值。
- 当测量最后一个电池的电压，5V的信号具有40V的共模电压（如图14-28所示）。

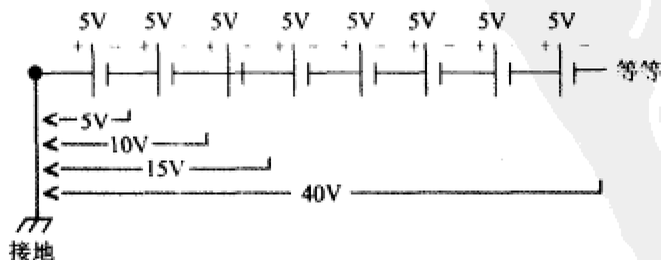


图14-28　共模电压

增益 信号被放大的因数，有时用分贝（dB）表示。一个简单的降低系统噪声影响的方法，就是在信号发送端把信号进行放大，使模拟量信号电平高于噪声电平，避免导线中的噪声的干扰。

图形用户界面（GUI） 一种直观、使用方便的信息交换方式，可以借助图形屏幕显示构成计算机程序。GUI 可以看成是指令的前面板，或者是联系计算机程序的界面。

接地 在数据采集系统中，有两种接地方式。系统接地或电源接地用来提供电流流动的通路，模拟接地或信号接地为信号调节器和转换器提供参考电压。一般模拟接地连接中不允许有电流流过。模拟接地也不允许相互之间，或与电源地之间连接，因为这会形成一条电流流通的环路。 401

输入/输出（I/O） 通过通信面板、操作接口设备和/或数据采集和控制接口，传送信息到计算机系统或接收来自计算机系统的信号。

隔离 测量信号的过程中，没有直接的电气连接（如图 14-29 所示）。隔离允许信号从信号源到测量设备的传输过程中，没有电气或物理的连接。利用光学或调制技术，去除高共模电压，隔断接地环路，期望的差分电压通过并被测量。隔离还可以提供重要的保护功能，避免高电压源如动力线、闪电和高压设备的损害。

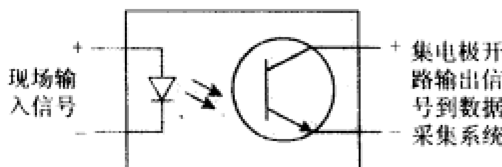


图 14-29 光隔离

多任务 可以同时运行多个进程的操作系统。

噪声 由其他信号产生的不希望的干扰。常见的噪声源包括：动力线（60Hz）、荧光灯和大的电机。滤波器可以滤除在特定频率范围内的不需要的噪声信号。

实时 在过程或系统中，数据按照需要及时处理，而不是被累积起来以后处理。

分辨率 可以被测量系统检测的最小信号增益。分辨率可以认为是一个量可被测量的接近程度。想象一个 1 英尺的直尺，如果上面仅有的刻度是英寸，那么分辨率就是 1 英寸；如果刻度是 1/4 英寸，那么分辨率就是 1/4 英寸。我们测量任何物体最接近状态也是 1/4 英寸。分辨率可以用分贝、比率和满刻度的百分比表示。例如，一个系统的分辨率可以是 12 比特、1/4096 和满刻度的 0.0244%。

采样保持放大器 设备对输入电压快速"抽点"采样，并且在 A/D 转换之前一直保持，这样可以防止 A/D 转换时测量信号改变和测量不准确。大多数快速（kHz）A/D 转换器，都采用采样保持放大器。同步采样和保持：

- 同时超过一个通道被采样。
- 每个通道都有自己的采样保持放大器。
- 全局保持所有的通道信号都被保持。
- A/D 转换器一次只对一个通道进行转换。
- 用户可以对同时从不同的传感器获得的采样值进行比较。

保存间隔或保存区间 规定数据保存的时间间隔（如图 14-30 所示）。保存间隔注意下面的内容： 402

- 可以是一常数。
- 可以随时间，或随着特定测量函数改变。
- 在许多数据巡回检测应用中，固定的采样频率将产生太多的数据，变采样速率只在发生改变时才对数据进行保存。

跟踪保持（T/H） 跟踪模拟量电压信号并保持的电路。

传输速率 在进行软件初始化和设置操作后，数据从信号源到目的地的传输的速率，以字节测量。最大传输速率由硬件水平决定。

触发 用来启动和/或停止数据采集系统。系统一旦获得启动信号（如一个触发信号），就开始测量并保存数据，一直到有停止信号才停止运行。

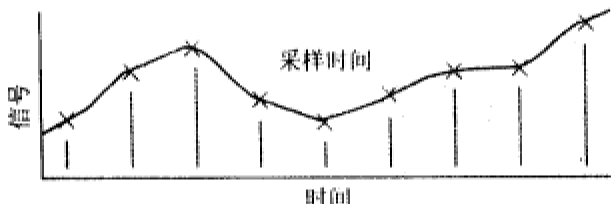


图 14-30 保存间隔

连接传感器与数据采集系统的电缆有多种类型。导线和电缆就如同天线一样，能接收外部环境的噪声。传感器和信号离得越远，越要注意选择电缆的配置。无屏蔽导线或带形电缆价格便宜，对于高电平信号传输较好，传输距离适中。但是对于低电平信号或者是传输距离很远时，要考虑使用屏蔽电缆或双绞线电缆。如果信号的带宽超过 100kHz，就必须使用同轴电缆，也可以使用光纤链路。光纤能够抗干扰，因为数据在链路上以光的形式传输。

另一个有用的减少噪声的方法是利用差动测量。因为采用差动测量时，从信号源到测量系统的 + 端信号线和 - 端信号线都拾取了相同的噪声。采用差动测量的输入，可以去除在两根导线上产生的噪声电压。 403

思考题

1. 列写出工业中 3 种过程处理类型。
2. 说出用于下列每一种应用的过程处理类型。
   a. 混合各种配料制造化学产品 b. 装配电视机 c. 电子板的大量生产
3. 说出要发展自动化控制机器和过程的 3 种理由。
4. 比较单机、集中和分布式控制系统。
5. 说出“过程控制系统”的含义。
6. 说出下面用于过程系统中的每一个部分的功能：
   a. 传感器 b. 人机界面 c. 信号调理 d. 执行元件 e. 控制器
7. 比较开环控制系统和闭环控制系统。
8. 根据提供的控制类型列写出控制器的 4 种类别。
9. 给出通/断控制器的运行次序。
10. 解释比例控制器是如何避免通/断控制器出现振荡的。
11. 比例控制器产生了什么样的过程偏差或偏移？
12. 积分控制是如何消除偏差的？
13. 控制器的微分作用是什么？
14. 列写出影响 PID 控制器输出响应的系统偏差信号 3 种因素。
15. 为什么 PID 控制器对于每一个控制过程都要进行常规调整？
16. 比较 PID 控制器的手动、自动和智能调整方法。
17. 给“数据采集”下个定义。 404
18. 概述一个典型的数据采集控制系统是如何工作的。
19. 解释下面与数据采集系统相关的每一个设备的功能：
   a. 变送器 b. 模拟多路复用器 c. 放大器 d. 信号调节器 e. A/D 转换器

20. 解释下面应用于数据采集系统运行的每一个单词的含义。
   a. 传感器校正　b. 漂浮信号　c. 增益　d. 隔离　e. 实时
   f. 分辨率　g. 保存间隔　h. 传输速率

习题

1. 给出一个开环控制系统的例子。解释为什么它是开环的，并找出控制器和末级控制元件。
2. 给出一个闭环控制系统的例子。解释为什么它是闭环的，并找出传感器、控制器和末级控制元件。
3. 如果死区范围太窄，通/断控制器将如何响应？
4. 一个家用通/断控制的加热系统，如果把死区范围扩大将会产生什么影响？
5. a. 一个温度控制器的带宽 5%，设定值为 500 °F，计算出它的比例区域。
   b. 计算出上限和下限。超出上下限，控制器就只能全接通或全关断。
   c. 计算比例增益：

$$增益 = \frac{100}{比例带宽}$$

6. 如果过程增益出现大的偏差，会产生什么影响？
7. 写出输入信号采用 4mA ~ 20mA 电流回路而不是 0 ~ 5V 电压信号的优点。
8. 怎样把数据采集系统应用于石油化工测量站？它们的主要目的是测量和流量控制。 405
9. 如果在应用中出现噪声问题，如何减轻噪声问题？

第 15 章　计算机控制系统

学习目标：

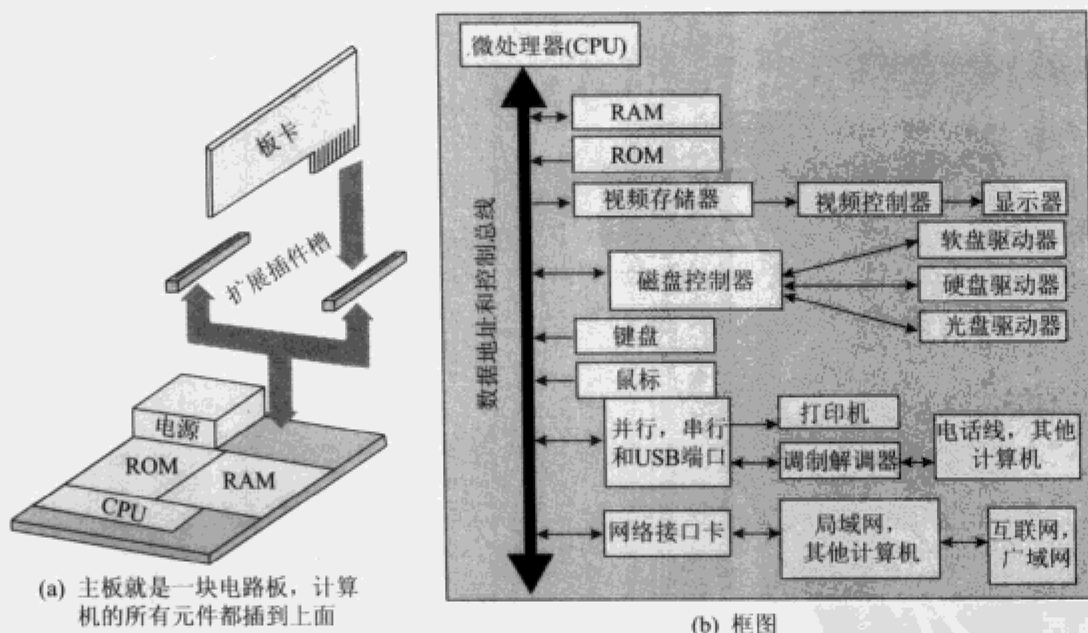
- ❑ 探讨计算机操作系统的工作原理。
- ❑ 解释工作单元的工作原理。
- ❑ 比较计算机之间的通信方式。
- ❑ 探讨计算机数字控制的工作原理。
- ❑ 探讨机器人计算机控制的工作原理。

因为个人计算机（PC）在工业中使用非常普遍，本章对重要的计算机操作做了简单的讨论，介绍了机器接口电路和协议标准，还介绍了计算机数字控制和机器人计算机控制。

15.1　计算机的基本原理

计算机控制的制造系统越来越多，因此理解计算机的基本工作原理是很重要的。计算机，特别是个人计算机，被认为是对机器控制和过程控制非常有用的设备。输入/输出卡，可以直接插到现今大多数个人计算机的扩展插件槽内。这些板卡可以实现数据采集、数据转换和电源控制等功能。配备特殊软件包和输入/输出板卡的个人计算机，在功能上可以赶上许多现今常用的 PLC。

所有的计算机都由两个基本部分组成：硬件和软件。计算机的*硬件*（如图 15-1 所示）包含了组成计算机系统的物理元件：



(a) 主板就是一块电路板，计算机的所有元件都插到上面

(b) 框图

图 15-1　计算机硬件

❑ 电源

把 120V 交流电压，转换为计算机系统需要的直流电压。它还对电压进行调节，去除

大多数电气系统中出现的峰值电压和浪涌。然而，并不是所有的电源都能较好的对电压进行调节，因此，计算机总是对一些大的电压波动比较敏感。电源的额定功率，是它能安全供电的瓦特数。

❑ 磁盘驱动器

计算机从盘中读取和写入数据。驱动器带动磁盘快速转动，有一个或多个磁头对数据进行读写。不同类型的盘对应了不同类型的盘驱动器。例如磁盘驱动器读取的是磁盘，光盘驱动器读取的是光盘。

❑ 硬盘驱动器

允许在不可移动的硬盘中存储和读取数据。硬盘驱动器是最重要的驱动器，它保存了计算机系统使用的大部分数据。“硬”这个词来源于包含在磁盘驱动器中磁片的类型，磁片由刚性玻璃或铝构成，不能弯曲，也不能像软盘一样任意移动。硬盘驱动器和软盘驱动器的工作原理很相似，都是有一个涡流磁盘，浮动磁头在磁盘上读/写数据。

❑ 主板

连接整个计算机系统的各个主要部分。主板包含了连接附加插件板的接口。通常主板包括：CPU、BIOS、存储器、大容量存储器接口、串行/并行端口和扩展插件槽。所有的控制器，都需要控制标准的外围设备，像显示器、键盘和磁盘驱动器。

❑ 微处理器（CPU）

解释计算机指令，并根据每一条指令执行相应的过程。主要功能包括：

- 处理系统数据
- 执行所有的数学运算
- 执行所有的逻辑运算
- 对系统进行控制，决定何时、怎样进行数据的传输

❑ 只读存储器（ROM）

在出厂前，存储器已经被编好程序，不能被用户修改。只能读取不能被写入，当电源断电时，内容保持不变。ROM 存储器包含了基本的输入/输出系统，可以借助它访问输入/输出设备。

407~408

❑ 随机存储器（RAM）

它是读/写存储器，用来存储计算机程序。当电源断电，它的内容会丢失。计算机 RAM 的容量越大，能运行的应用程序也就越大。

❑ 总线

计算机各个部分之间发送和接收信息，所需的一组二进制数据线或连接电缆。可以认为总线是数据在计算机内部传输的高速通路。当用在个人计算机上面，总线一般指的是内部总线，它连接计算机内部的各个部件到 CPU 和主存储器。扩展总线使扩展板能访问 CPU 和存储器。所有的总线都包含两个部分：地址总线和数据总线。数据总线传输实际数据，地址总线传输的是数据的地址信息。总线的大小用它的带宽表示，带宽非常重要，因为它决定了一次可以传输多少数据。例如，一个 16 位的总线可以传输 16 位的数据，而 32 位的总线可以传输 32 位的数据。

409

❑ 外围卡

使计算机能访问和连接输入、输出设备，如驱动器、打印机、显示器等外部设备。

❑ 扩展插件槽

用来将其他电路连接到主板的连接插件。

计算机软件能给计算机提供“生命”。软件常驻在硬件里面，可以把软件看成计算机程序，它仅仅是告诉计算机该怎样做的一系列指令。程序通常储存在一些大容量存储器中，如光盘或软盘，需要时把它装载到计算机的 RAM 中。

磁盘驱动器既是输入设备也是输出设备，因此它既可以向计算机传送信息，也可以存储来自计算机的信息。

软盘是一种磁存储器，它存储信息，以后可以恢复（如图15-2所示）。信息存储在磁道上面的扇区中。存储区域的设置，使计算机可以访问磁盘扇区并读取数据。

软盘上的小孔是写保护用的。当小孔打开时，不能写任何信息到磁盘上。现今使用的标准软盘是3.5英寸的高密度（HD）磁盘，它能在两面存储1.44M的数据，每一面有80个磁道，每一磁道有18个扇区。

所有的软盘在使用之前必须先格式化。许多（不是所有）软盘都被生产厂家预先进行了格式化。格式化使磁盘与正在使用的计算机兼容。一旦软盘插入计算机，驱动电机带动磁盘转动，步进电机使读/写磁头在磁盘不同的位置移动。指示灯指示磁盘驱动器是否运行。

410 软盘通过一个带有标记的窄口，插入软盘驱动器。然后软盘驱动器的门关闭，使软盘读磁头能读取信息。在使用软盘时，必须要小心，否则存储在软盘中的数据很容易被破坏。

- 不要触摸暴露的磁头缝或表面。
- 不要把软盘暴露在加热环境或强太阳光下。
- 不要把软盘暴露在常出现电磁或静电场区域的工业环境中。
- 只使用专门为磁盘设计的标签，这些标签在磁盘移动时，不会粘在磁盘驱动器上。
- 不要把软盘强制送入磁盘驱动器。
- 要对磁盘做备份，把母盘保存到安全的地方。

光盘是利用激光读写数据的存储媒介。一张光盘可以存储700M的信息，比软盘的容量大很多（如图15-3所示）。光盘有三种基本形式：

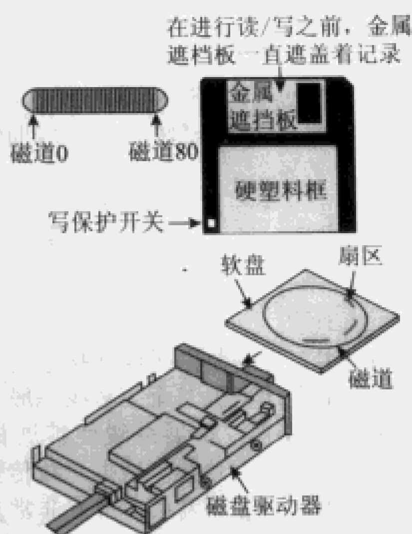


图15-2 软盘

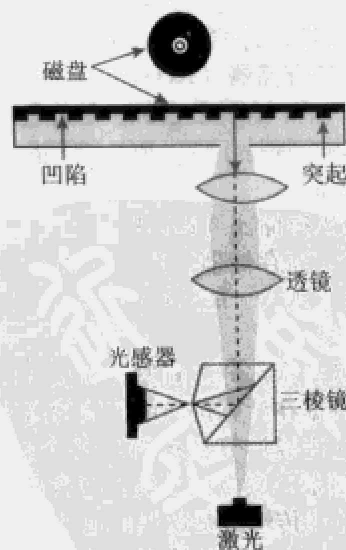


图 15-3

只读光盘（CD-ROM）。高密度盘片——只读存储器。

可记录光盘（CD-R）。可记录（数据一旦被写入光盘，就不能修改）。

可擦写光盘（CD-RW）。磁盘上原有的数据可以被擦除，也可以装入新的数据。

硬盘驱动器存储和恢复数据的速度比软盘更快，存储的信息也更多。硬盘的大容量存储系统，永久固定在计算机内部，可以同时存储大量程序。硬盘以磁性形式存储数据，通常把磁膜放置在铝或玻璃瓷盘片上。典型的硬盘驱动器包含了多个相互叠加在一起的磁盘

411 片，并置于密封的磁盘驱动器盒内（如图15-4所示）。

软盘上的内容可以重新下载到硬盘中，程序在硬盘上按要求执行，软盘作为备份。硬盘驱动器通常指的是C:盘（一般在驱动器名后面加冒号）。如果使用了一个软盘驱动器，它被称为A:盘，如果再使用第二个软盘，则被称为B:盘，其他的驱动器称为D:盘、E:盘等。

软件的分类主要有两种：系统软件和应用软件。系统软件提供使用户与计算机进行交换信息的程序，实现对磁盘、打印机和计算机使用的其他设备的操作。操作系统软件包含了命令和程序，使用户能与计算机进行交换信息。系统软件可以认为是在硬件设备和应用程序或用户之间的翻译器（如图15-5所示）。应用软件是指写入到计算机的、完成特定应用的程序，例如字处理软件。软件市场中的大部分软件都是应用软件。如果要计算机运行应用软件，必须先将系统软件装入计算机存储器中。

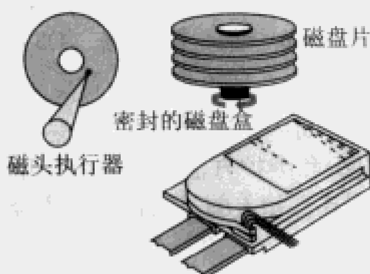


图15-4　硬盘驱动器单元

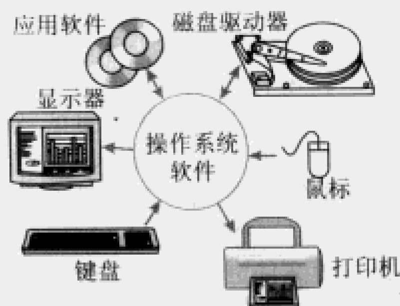


图15-5　操作系统软件就像是硬件设备和应用程序之间的翻译器

微软DOS系统（磁盘操作系统）是一种命令行式的用户界面。MS-DOS 1.0 在1981年发布，应用于IBM计算机上。MS-DOS的最新版本是1994年发布的MS-DOS 6.22。现在MS-DOS已经不常使用，但是不同版本的Windows系统都可以对其访问。计算机的操作系统（OS）完成基本的工作，例如从键盘和鼠标接收信息、发送信息到监视器、存储信息到硬盘和控制外围设备等。它为所有软件提供了一个命令平台，完成以下功能：

- 控制输入和输出操作
- 解释和执行由键盘和鼠标输入的命令
- 读、写和编辑文件
- 备份文件
- 组织磁盘
- 文件管理
- 显示磁盘内容
- 查看错误信息

文件就是将数据的集合存储在同一名字下。当将信息存储在磁盘上时，文件被保存或写到磁盘上。当要调用信息时，可以对磁盘进行读操作或将数据下载到计算机的存储器。可以对文件进行移动、复制、重命名和删除。一些重要的文件类型包括：

- **可执行文件**

  可被CPU执行的一串指令。
- **数据文件**

  信息。
- **文本文件**

  一系列的字符，如字母、数字、标点符号和字间距等。
- **图形文件**

  可转换为数字码的图形。

磁盘被格式化以后，对文件进行读或写的过程，涉及到操作系统软件、计算机的基本输入/输出系统（BIOS）和磁盘驱动器本身。BIOS 是计算机硬件和软件程序的连接纽带，它永久的存储在计算机的内存中，是 CPU 和与之连接的其他设备的翻译器。

当计算机第一次启动时，BIOS 使计算机检测软驱和光驱，检测是否存在包含操作系统的磁盘。如果有，计算机就自动装载操作系统。把操作系统装载到计算机的过程被称为引导系统。如果计算机中不存在包含操作系统的软盘，BIOS 就会检测硬盘驱动器（C:盘），看是否有操作系统。如果没有操作系统存在（或没有硬盘驱动器），就会显示错误信息（如图 15-6 所示）。

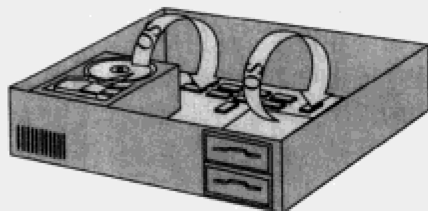
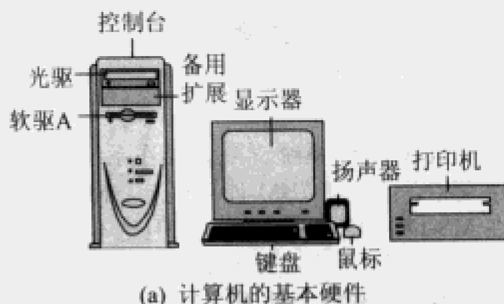
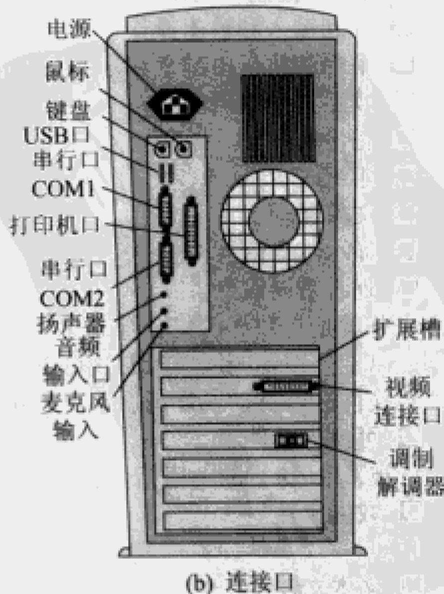


图 15-6　装载操作系统（OS）到存储器

“目录”和“文件夹”这两个词可以互换。文件夹或目录，是组织软盘和硬盘中所有的程序和文件的一种简单方法。由于磁盘存储了大量的数据，因此把数据分开存储于具有单独命名的区域内是非常有用的。每一个区域用来存储一些相关的文件（如字处理文件）。这种设置提供了简单的硬盘组织和维护方法，这也就是用文件管理器（如资源管理器）查看驱动器内容时所能看到的样式。相关的一组信息被集合到一起形成文件，相关的文件进一步集合到一起，形成目录（或文件夹）。目录也被分级排列成所谓的指令树的形式，其中主干由根目录确定（如用 C:\表示，一般是计算机上面的硬盘驱动器）。每个主目录是根目录的子目录，每个主目录也许还有它自己的子目录。包含子目录的目录被称为是子目录的母体。如果不理解目录结构，在上传或下载文件时，可能会在硬盘的不恰当区域删除文件或保存文件。



(a) 计算机的基本硬件



(b) 连接口

图 15-7　计算机的外围连接

微软视窗操作系统（Microsoft Windows®）是 MS-DOS 系统基于图形的接口界面。它用图形映像代替了在 DOS 中运行的指令，用选择图形（图标）的方式代替了键入命令。Windows 系统允许执行多个任务。没有哪个计算机可以真正一次处理两件事情，但是可以通过在一个程序和另一个程序之间快速的前后移动来实现。Windows 系统中的程序共享 CPU 资源，当计算机没有被用户控制时，控制权不时的前后传递。Windows 允许并行和同时处理多个特殊的应用操作。

端口是凸显在计算机机箱后面的连接设备（如图 15-7 所示）。端口实际上是扩展卡的后端，它们提供计算机与外围设备连接的电气接口。端口通常包含了一定数量的引脚，保证可靠的电缆连接。端口可分为凸的和凹的（引针和引脚匹配），也可以分为串行口（如连接鼠标的口）和并行口（如连接打印机的口）。串行口在单独、单向线上一次只发送 1bit 的数

据；并行口穿过8位并行数据线，可同时发送多位数据。因为串行口比较简单，它可以用于计算机和其他任何可能设备之间的数据通信。串行口常常指的是RS-232口（美国电子工业协会命名）。并行口常常称为Centronics端口。端口（COM）就属于串口，打印机接口或LPT口（行式打印机）属于并行口。

通用串行总线（USB）属于外围总线标准，它可以把大量外围设备连接到计算机（如图15-8所示）。由于这种端口的传输速度快和可扩展，它正迅速的取代串行口和并行口。USB1.1口的传输速度大约是12Mb/s，USB2.0最大传输速度可达480Mb/s。数码相机、鼠标、键盘、调制解调器、打印机、游戏杆和一些扫描仪都是常用的USB设备。可以用菊花链的形式将127个设备连接到一个单独的USB口。

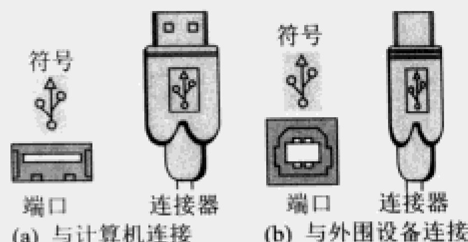


图15-8 通用串行总线（USB）属于外围总线标准，可以把大量外围设备连接到计算机

一个计算机具有多个连接设备的端口。确认需要通过什么类型的端口把设备与计算机相连是很重要的。当计算机的端口与设备的连接器不匹配时，需要增加适配器。其他的计算机端口包括：

IEEE 1394口 这种端口也被称为火线口，是USB接口的竞争对手，特别是在视频和多媒体领域。它可以连接63个设备。现今它能将数字卫星系统连接到计算机。

电话接口 许多计算机用RJ-11（Registered Jack-11）作为电话接口。这些端口可以使计算机通过拨号直接连接到互联网和传真文件系统。

414

以太网接口 被称为RJ-45，除了比电话接口大一些外，其他的都很相似。通过此端口计算机可以连接到其他网络，进行高速通信。以太网端口，是常用的连接过程控制设备到计算机的方法之一，如计算机与输入/输出模块的连接。使用以太网端口，可以将输入/输出模块放置在靠近过程设备的地方，这样计算机就可以放在比较好的环境中。

PS/2端口 针对鼠标和键盘而设计的。

音频端口 音频口取决于计算机上面的声卡。可用的音频口有麦克风、扬声器、音频-入和音频-出。

视频端口 这种端口把来自显卡的数字量信号转换为传送给显示器的模拟量信号。平板显示又利用这种端口把模拟量转换回数字量。

小型计算机系统接口（SCSI） 发音scuzzy，SCSI接口是一种并行接口，它比标准的串行和并行接口有更快的数据传输速率（达到80Mb/s）。而且，一个单独的SCSI接口可以连接很多设备。

15.2 计算机集成制造

如今自动控制正向集中控制的方向发展。系统操作员借助个人计算机，只需用鼠标轻轻单击就能实现对整个系统的控制。用计算机监视和控制的联网PLC系统如图15-9所示。

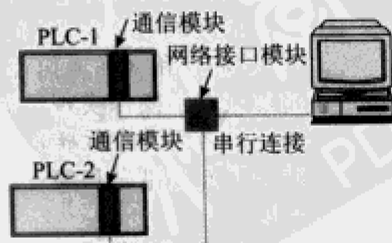


图15-9 用计算机进行监视和控制的联网PLC系统

计算机集成制造（CIM）系统提供独立的具有数据通信功能并相互兼容的计算机，使多个独立设备集成到一个独立的系统中。一般来说，CIM需要四种级别的计算机集成：单元级、区域级、车间级和设备级（如图15-10所示）。每一级在它的责任范围内，都有确定的任务：

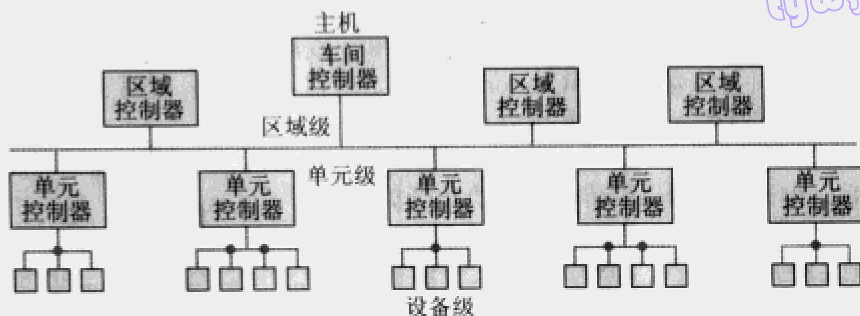


图 15-10　计算机集成制造系统的控制级

- **车间控制器**

415

用于完成采购、会计、材料管理、资源计划和生成报表等任务。

- **区域控制器**

用于完成机器和设备管理、维护调度、材料处理和调度、机助仿真等任务。

- **单元控制器**

用于控制设备和采集数据。

- **设备控制器**

用于直接控制那些生产或处理产品的设备。

工作单元可以被定义为：一组机器设备集结到一起，完成制造系统的特定任务。与机器相关的典型工作单元如图 15-11 所示。计算机或单元控制器，是各个部分之间的通信连接器。PLC 与单元控制器之间的主要区别在于单元控制器编程的编程语言不同。PLC 的编程语言简单，需要很少的编程知识；而单元控制器需要更多的编程知识。基于这种想法，PLC 生产厂家开发的 PLC 软件都有更加强大的功能。这种努力的最新成果就是一种基于计算机的控制器，被称为软 PLC。Allen-Bradley 公司的 ControlLogix 控制器，就是这种类型的 PLC。这种基于 Windows 操作系统的软 PLC 控制系统，结合了 PLC、编程终端、操作界面和数据采集的功能。它的优点有：

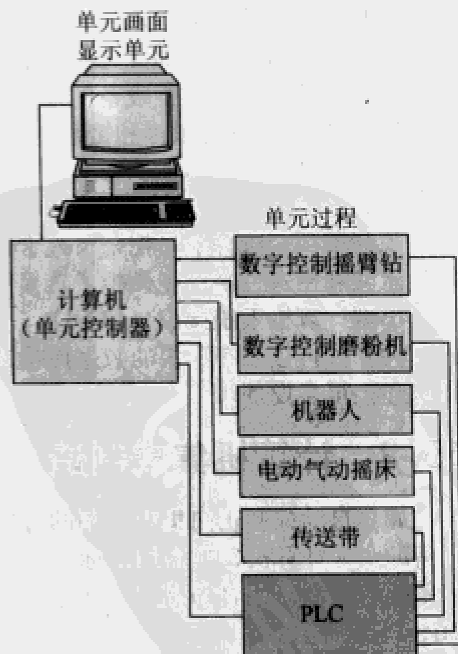


图 15-11　与机器相关的工作单元

- ❑ 用先进易用的基于 Windows 系统的图形流程图编程工具代替了继电器式梯形图编程。

416

- ❑ PLC 的功能、基于 PC 的编程、人机界面、输入/输出和网络通信都能在一个基于 PC 的数据库中操作实现，而且可以使用相同的标签名称，从而避免了使用多个数据库，减少了人为错误和编程错误，加快了编程速度。
- ❑ 通过在同一计算机上安装扫描卡，访问不同系列的输入/输出模块。

工业计算机的出现，使得单元控制在工业中的大量使用成为可能。越来越多的单元控制程序在计算机硬件平台上运行。单元控制，是指一系列的自动化可编程控制机器

(PLC、机器人等) 组合到一起共同完成所有的制造或与过程相关的任务。单元控制器的功能，包括它的通信和信息处理功能，调整和监视单元内部机器控制的运行。即使设备相互之间不能通信，它们也必须能与单元控制器进行通信。单元控制器必须能上传/下载程序、交换变量信息、启/停设备和监视每台设备的运行。

如今单元控制还涉及那些影响生产线的因素。以装配生产线为例，如果生产的部件需要被检测，由机器人取下部件，把它放在被检测的位置。根据检测的结果，把它放回生产线或改变位置放置。如果部件不合格，需要把这个信息向上或向下传送到它能起作用的地方。还要重新改变位置进行放置，用清单描述出问题的所在，这样部件就不会被再次检测，而是对其修理、调整并放回生产线。这个过程需要通信链路，以避免当部件信息没有被更新时，生产出不合格的产品。

15.3 数据通信

数据通信指的是指基于微处理器的系统之间或与其他设备之间，通过不同的方式进行对话。最开始，PLC 配有数据通信系统，但是很明显，计算机和数字控制的机器以及机器人控制需要连接到 PLC，PLC 反过来通过工厂局域网连接到主机和其他设备。

PLC 之间或 PLC 与计算机之间通信的应用，已经非常普遍。局域网是通信网络的主干。**局域网**（LAN）的基本功能是提供 PLC 之间或 PLC 与计算机之间的通信。在工业中，传输媒介最常用的是同轴电缆或对噪声高免疫的光纤。数据在通信线中的传输速率，取决于一次能传送的二进制信息位数。信息的传输用每秒多少位来测量。PLC 可以通过串行连接到局域网。该网络允许 PLC 从存储器复制输出信息到局域网链路，也可以从局域网链路复制信息到 PLC 的存储器。

417

局域网有三种基本拓扑（通信网络的物理布局或结构）：星形、总线形和环形（如图 15-12 所示）。设备与传输媒介的连接点称为节点或站。

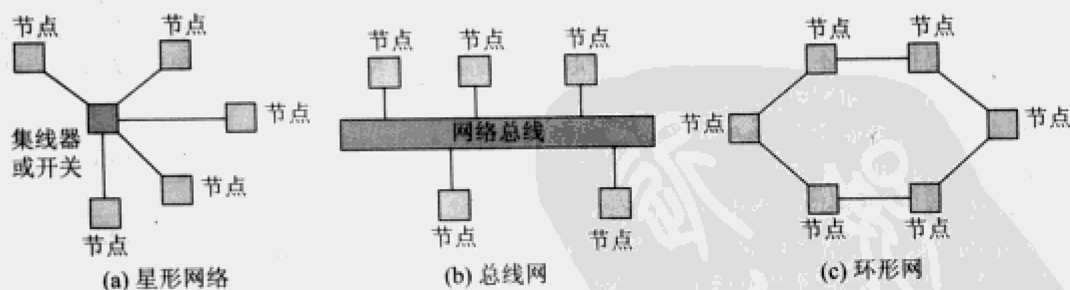


图 15-12　网络拓扑

在**星形网络**中，中央集中控制设备或集线器连接多个节点。这种结构允许中央控制设备和每个节点之间进行双向通信。由于中央控制设备控制所有的通信，所有的数据传输都是在中央控制设备和节点之间进行。所有的数据都被发送到集线器，然后集线器再把数据发送到正确的节点。这种星形网存在一个问题，如果集线器出现故障，整个局域网就会瘫痪。这种类型的网络最适用于信息主要在主控制器和远程 PLC 之间传输的系统。然而，如果大部分的通信是在 PLC 与 PLC 之间，运行速度就会受影响。而且，这种星形网络可以利用充足的通信导线，把所有的远程 PLC 连接到中央位置。

PLC 网络最先采用星形拓扑结构，这种结构包含了一个带有多端口的主机，每一个端口都连接到 PLC 的编程口。图 15-13 所示为这种配置结构。网络控制器可以是计算机、PLC 或另外的智能主机。

在总线网络中，每个节点都连接到中央总线。当节点向网络发送消息时，每个消息有一个目的站或节点号。当消息在总线上传输时，每个节点都监听自己的节点辨认号，并只接收发送到该号码的信息。控制类型可以是集中控制，也可以是节点之间的分散控制。不需要信息经过中心节点，任意两个节点之间都可以进行通信。这种结构是一种并行通信系统，总线只是用来连接PLC的通信电缆。

418

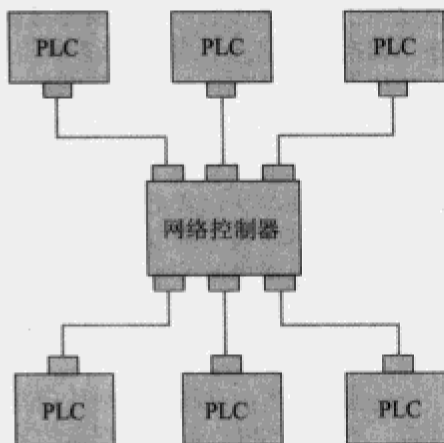


图 15-13 PLC 星形网络

总线就是传输数据和/或电源的导线集合。总线网络在分布式控制系统中非常有用，因为每个站或节点都具有相同的独立控制能力，并且能在任何时刻进行数据交换。总线网络的另一个优势，就是通过对系统进行最少的重新配置，就可以在网络中增加或去除站。这种网络的主要缺点，就是所有的节点都依靠主干总线，主干线故障会影响到多个节点。

在环形网络中，每个节点与其他节点顺序相连形成环路。信息在环路中，从一个节点向另一个节点传送。信息被赋予特定的站或节点号，每个节点监听自己的节点辨识号。控制类型可以是集中控制或是分散控制。这种系统类型非常适合于令牌传输协议。令牌或位模式，被节点用来获得通信通道的控制。实现计算机集成制造的关键，是包含于控制过程中的所有控制器之间的通信能力。尽管存在通信标准，但是每个计算机系统可以使用不同的标准。而且，随着通信复杂性的增加，标准的完善性也要增加。

协议指的是一种预先规定好的约定，它规定了在两个或多个设备之间传输的信息格式和同步。只有对传输规则达成一致，才能进行通信。协议对于计算机，就相当于语言对于人，这本书是用英语写的，要理解它就必须能阅读英语。类似地，网络中的两个设备要能成功地通信，它们就必须遵循相同的协议。有多种标准来规定一个协议，对于计算机集成制造系统，要把所有不同的设备连接到一起，就必须使用共同的协议。如果协议不同，就需增加附加的硬件和编程。

原来通信网络通常是专有网络，它的设计只符合特定卖家的标准，用户只能买单一供应商的产品，因此选择性方面受限。如今由工业协会发展的基于国际标准的开放性网络系统是大势所趋，Allen-Bradley 公司提供的开放性网络包括 DeviceNet、ControlNet 和 EtherNet/IP。生产厂家和第三供应商也出售接口设备和软件，把一种网络访问结构转换为另一种网络结构。如果接口设备只有转换网络访问功能，那么这种设备称为网桥；如果也能调整数据类型或具有数据传输控制器功能，这种接口设备称为网关（如图 15-14 所示）。

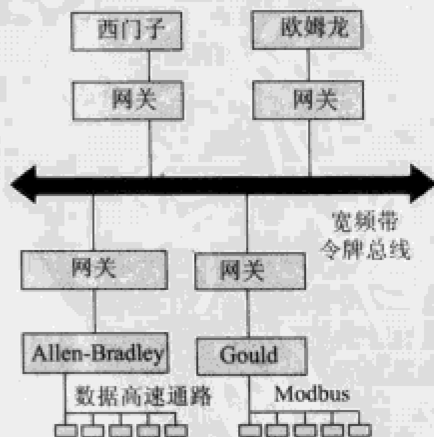


图 15-14 网络互联

局域网协议就是局域网采用的一种确定从通信信道接收和传输数据的标准方法。它实际上是一种局域网在数据高速通路上“指挥交通”的方式。两种基本的网络通信访问机制方式是：主－从方式和对等方式。主－从系统（如图 15-15 所示）的特征是在同一总线上由一个主机控制所有的网络设备之间的通信。在给网络主设备

419

编程时，通常都给每个从站设备单独分配一个地址。在从站之间传输的信息，必须先发送到网络主站，然后再由主站把信息发送到目的从站。

PLC可以采用多种访问方式在网络上传输信息。在主－从协议中最常用的访问方式是轮询。主站按次序访问每一个站（从站），检查其是否有数据要传输。主站发送消息给特定从站，并等待固定的时间使从站响应。从站响应时，或者发送数据或者发给主站一个特定消息表明它没有数据可发送。如果从站在分配的时间内没有响应，主站就认为它已经停止并继续轮询其他的从站。

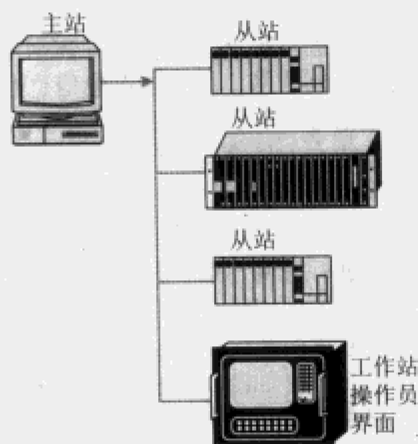
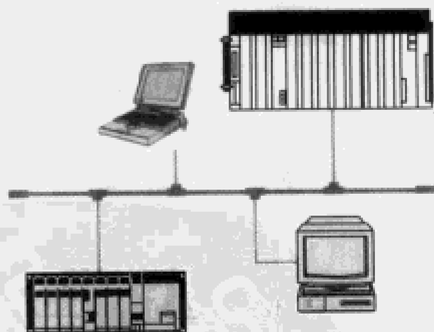


图15-15　主－从访问方式

在对等系统中，每个网络设备都能请求使用并控制网络，进而在网络上传输信息或者从其他网络设备中请求信息。因为网络的控制可以看作令牌从一个设备单元向另一个设备单元传输，这种类型的网络通信方案常常被称为令牌传输系统。在令牌传输系统中，一次只能有一个站响应，节点或站必须拥有了令牌后，才有权使用线路。令牌一直在站之间循环传输，直到有站想使用线路，节点就"抓住"令牌并使用线路。节点只有在它的信息传输期间才拥有令牌，一旦节点传输结束，令牌又重新自动的在局域网中各节点之间循环。令牌传输方式如图15-16所示。 420

在公共总线网络配置中，使用令牌传输技术，每个站通过一个地址进行辨认。当网络运行时，令牌从一个站到另一个站顺序传输。传输令牌的节点知道将要接收令牌节点的地址。网络以一个或多个数据包的形式传输信息，数据包中包括源地址、目的地址和控制信息。每个节点如果有需要，就接收这个信息并使用它。节点增加了附加的信息，以新的数据包发送出去。


图15-16　对等或令牌传输访问方式

有两种基本的数据传输方式（如图15-17所示）。一种是并行传输，可以同时传输多位；另一种是串行传输，一次只能传输一位信息。并行数据通信在低于30英尺的距离内，效果比较好。串行数据通信，只需两根线并能传输相当远的距离。串行通信中每个数据字，必须用开始位、数据位（包含了传输信息）和一个停止位隔开。常常会增加一个奇偶校验位，提供检错能力。

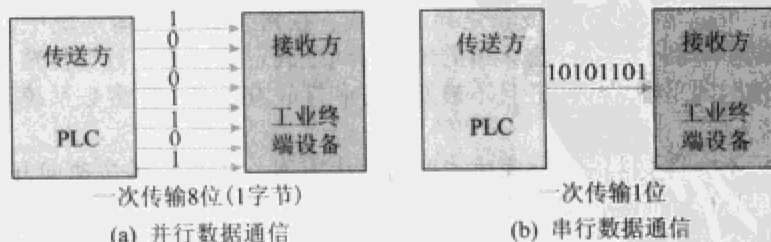


图15-17　并行和串行传输

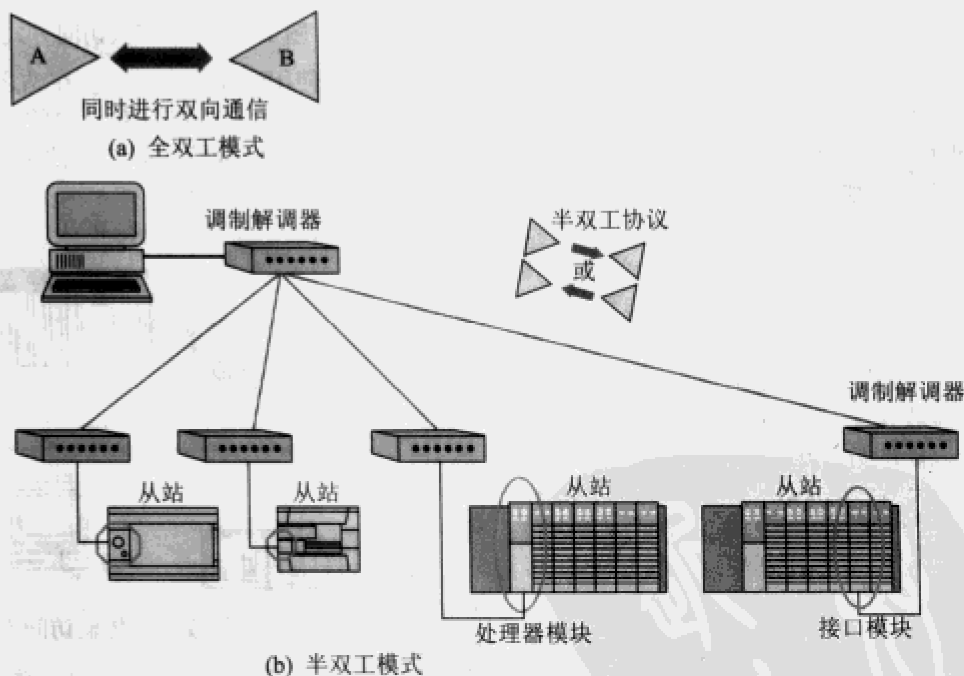
调制解调器使任意的两个设备通过单独的一对专用导线或是电话线进行通信（如图15-18

所示)。一对调制解调器可用来发送和接收数字量信号，它通过不同的音调表示逻辑 1 信号和逻辑 0 信号。如果在两个距离非常远的点之间进行数据传输，或者数据传输采用的是公共电话网，那么数据位信息必须转换为音频信号，因为以电压信号表示的数据位，很可能大大地衰弱了，以致数据信号不能被读取。调制解调器，用于把数据位信号转换为音频信号。它把信号在发送方调制，在接收方进行解调，把音频信号又转换为数据位信号。单词“modem”由“modulate”的前两个字母和“demodulate”的前三个字母组成。



图 15-18 利用一对调制解调器的数据通信

通过调制解调器进行数据传输有多种标准。*双工模式*指的是在两个设备之间进行双向数据通信（如图 15-19 所示）。全双工传输模式允许在两端同时进行数据传输。半双工传输模式使用一个开关，使信息在某一时刻只能在一端发送。



(a) 全双工模式

(b) 半双工模式

图 15-19 双工模式

数据通信，可以是同步方式或者是异步方式。在同步数据传输系统中，一旦传输开始，从一个数据字节到下一个数据字节的传输时间是确定的。异步数据传输，指的是数据字节之间的传输时间是随机的，并且不被发送方和接收方预知，大多数系统采用的是异步数据通信。

421

将来，工厂不会再受限制于一家生产商的解决方案。工业市场快速向开放系统发展，不论是硬件还是软件方面。PLC 越来越小型化，这些小型 PLC 和更多的分布式输入/输出模块变得越来越受欢迎。工业自动化系统的重要环节，涉及输入/输出处理、大量传感器输入信号的测量和调理、记录、显示、训练、报警和基于那些传感器输入提供的测量值的控制。末级控制输出通常是电磁阀、阀门、加热器和其他的执行元件，这些元件是通过与输入信号相关的控制算法进行操作的。存取实时产品数据，是获得最佳生产性能的重要因

素，通常是历史生产数据的一个综合数据库。图 15-20 所示为多个级别通信构成的一个综合数据库。

PLC 之间和 PLC 与计算机之间的通信，可以采取多种类型的网络。通信网络可以用来监控、数据采集、监视设备、处理参数和下载/上传程序等。以下是对这些网络特征的简单总结。

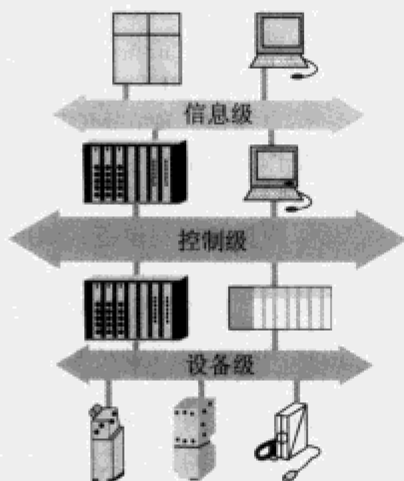


图 15-20　多级存取实时产品信息

- **数据高速通路**：DH-485 是专门用于 Allen-Bradley SLC-500 处理器的通信网络。它最大支持连接 32 个设备和 4000 英尺的网络距离。同一制造商生产的不同族群的 PLC，可能由于使用不同的协议，它们之间不能通信。例如，Allen-Bradley PLC-5（数据高速脉冲网络）需要增加网桥才能与 Allen-Bradley SLC-500（数据高速通路 DH-485 网络）处理器通信，这是因为它们的协议不相同。
- **RS-232**：RS-232 是串行二进制通信的电子工业协会（EIA）标准。通过 RS-232 端口，只需合适的电缆就可以把计算机与 Control-Logix 或 SLC-500 控制器直接相连。计算机与 PLC 之间连接的最大允许距离为 50 英尺，数据通信速度比其他网络要慢。
- **数据高速脉冲网络**：DH+是一种局域信息网络，它为远程编程和存取、传输数据而设计，是 Allen-Bradley PLC-5 控制器的主要通信网络，最大可连接 64 个设备，最远网络距离为10 000英尺。
- **ControlNet**：ControlNet 是一种开放、高速和确定性网络，它能在网络中传输实时 I/O 数据、控制器之间的互锁数据以及重复性的非实时数据（如数据监控）。确定性能准确预知数据何时传输；可重复性保证传输时间的固定，不会因为增加或去除网络设备而产生影响。网络技术规范和协议是开放的，这就意味着用户不必为了把设备连接到该系统而去购买其他任何的硬件、软件或许可证。
- **设备网**（DeviceNet）：随着 PLC 的功能越来越强大，它们需要在更大的区域范围内控制产品生产。因此，有时把每个传感器和执行元件，直接单独地连接到 I/O 模块是不实际的。设备网是由开放设备网销售商协会（ODVA）负责的开放式网络。通过开放性网络，设备网允许把设备直接与车间级控制器相连，而不必通过硬接线把每个设备连接到I/O模块。这种直接连接减少了导线的费用并方便安装。
- **以太网**（EherNet）：以太网是一种流行的全厂范围内的信息网络，它为实现计算机和其他以太网设备之间的高速数据交换而设计。以太网的优势在于可以采用大量的产品进行长距离的通信传输，而这些产品有众多的制造厂家供应。以太网工业协议（EtherNet/IP）是一种开放的工业网络标准，它支持实时 I/O 通信和数据交换（如图 15-21 所示）。这种协议的出现是由于迫切需要将以太网用于控制领域。以太网伴随一系列的规则，支配它的基本运行功能。以太网中，只要网络没有被使用，任何计算机都可以在网络上发送数据。如果两个计算机想要同

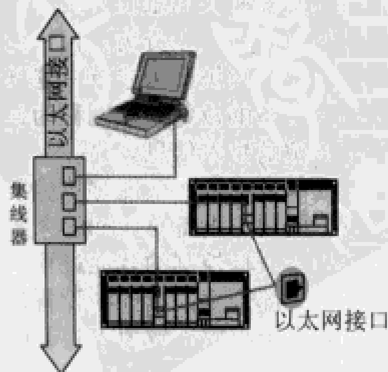


图 15-21　EtherNet/IP 网络

422

时发送数据就会发生冲突，两个计算机发现冲突存在会停止发送。

423

- **MODBUS**：MODBUS/TCP 是从 MODBUS 家族中派生出来的。MODBUS 是一种简单的、对销售商中立的通信协议，用来监控自动化设备。特别地，这个协议覆盖了 MODBUS 在企业网或使用传输控制/互联网协议（TCP/IP）互联网环境下的信息通信。现在 MODBUS 协议普遍的用于以太网，连接 PLC、输入/输出模块或是作为其他简单的现场总线设备或 I/O 网络的网关。
- **现场总线**（Fieldbus）：现场总线是一种全数字、串行双向通信系统，它直接连接了测量和控制设备，如传感器、执行元件和控制器等。在工厂网络的底层，对于在过程控制和制造自动化系统中使用的仪表设备，现场总线担当局域网一样的功能，并且有潜在的把控制过程沿网络分配的能力。现场总线是一种开放的非专有协议，任何厂家都可以应用它。
- **Profibus-DP**：Profibus-DP 是广泛适用于制造业和自动化系统的开放式网络标准。它工作在现场设备层，如功率表、电机保护器、断路器和照明控制。它使 PLC 在分布式系统内，最大限度的发挥功能。

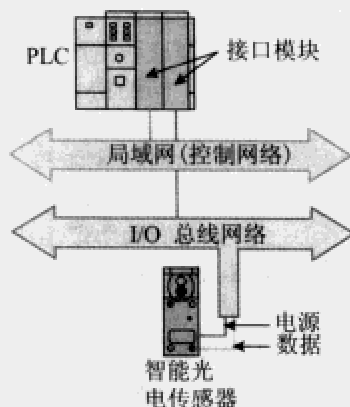


图 15-22　PLC、局域网和 I/O 总线网之间的典型连接方式

对现场设备进行分散控制和引入智能化的需求，产生了功能更强大的网络—I/O 总线网络。在 I/O 总线网络中，PLC 与现场输入/输出设备的通信模式类似于在局域网中监控 PLC 与其他单独的 PLC 进行的通信。I/O 总线网络的基本功能是共享数据，并给连接到网络上的设备提供电源。图 15-22 所示为 PLC、局域网和 I/O 总线网之间的典型连接方式。PLC 直接驱动现场设备，没有使用现场输入/输出模块，根据总线协议与现场输入/输出设备进行通信。

PLC 连接到 I/O 总线网，除了要使用 I/O 网络总线扫描器外，其他都类似于连接到远程输入/输出模块的方法。I/O 网络总线扫描器读/写现场设备的地址，并解码包含在网络信息包中的数据。连接到网络上的现场设备，通过嵌入微处理器或其他电路的形式，具有智能性。这些设备不仅交换通/断状态信息，还能对自己的运行状态进行诊断。例如，当透镜变脏时光电传感器就不能对边沿监测，可以通过网络监测在传感器检测物体失败之前，纠正这种状况。限位开关可以显示它的通断次数，这样就可以监视它是否达到运行寿命，而决定是否需要更换。

I/O 总线网可以分为两类：设备总线网和过程总线网。设备总线网连接低电平信号设备，如按钮开关和限位开关等，它们主要传输与通/断状态和运行状态相关的数据。设备总线网可以进一步分为：位－宽总线和字节－宽总线。包含了数字设备和少量模拟量设备的设备总线网被称为字节－宽总线网。这种网络一次能传输 50 或更多的字节数据。只连接数字设备的设备总线网称为位－宽总线网。位－宽总线网从简单的数字设备接收或传送少于 8 位的数据信息。

424

过程总线网一次能够传输几百个字节的数据。用于过程总线网的大部分设备是模拟量设备，而用于设备总线网的大部分是数字量设备。过程总线网连接高电平信号设备，如典型的用于过程控制的智能过程阀和流量计。由于传送的数据包很大，过程总线网的速度比较慢。大部分模拟量控制设备控制像流量和温度这样的物理量，响应都很慢。

有多个组织正致力于为 I/O 总线网确定协议标准。图 15-23 所示框图为现存的网络和协议标准。致力于确定过程总线标准的主要组织是 Fieldbus 基金会和 Profibus 行业组织。一些生产厂家都使自己的模拟量产品与 Profibus、Fieldbus 或其他的网络协议模式相兼容。

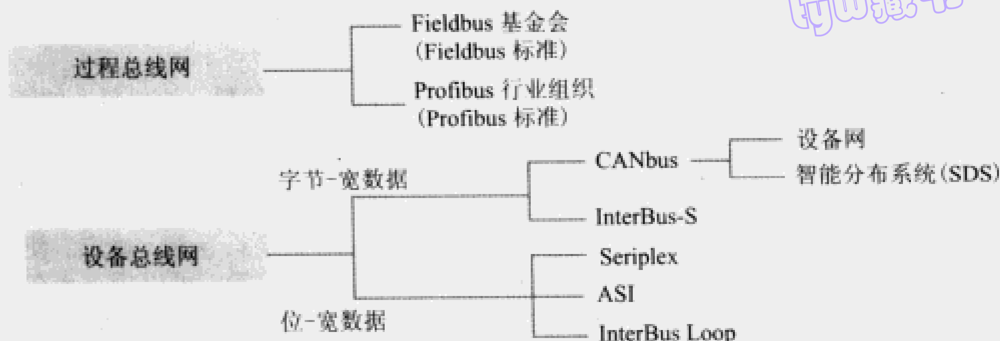


图 15-23 现存的网络和协议标准

虽然没有统一的设备网协议标准，但还是出现了几种由设备网制造商提供的协议，它们的技术规范比较实用。为了发展开放性网络结构，这些网络制造商或协会提供给现场输入/输出设备制造商技术规范。设备网就是适合于字节－宽设备网的一个标准，它最初来自 PLC 制造商 Allen-Bradley 公司，现在由一个独立的协会——开放设备网销售商协会（ODVA）提供。另一个是智能分布系统（SDS），它来自 Honeywell 公司。这两种设备网协议标准都是基于适用于工业自动化领域的控制领域网络总线（CANbus）的，它们把在市场上可获得的 CAN 芯片放入它的协议中。 425

15.4 计算机数字控制

PLC 控制系统不是唯一的用于机器控制的系统。如车床和磨床这样用来制造非常精密的机器部件的机床，它的控制系统常常结合 PLC 控制和计算机数字控制（CNC）。CNC 控制的车床系统，可以加工非常复杂和精密的部件。

概括地讲，数字控制（NC）是一种灵活的、通过数字量自动控制机器的方式。数字控制设备由连续通路或多个点驱动，通过程序数字化地调整机器移动运行，制造产品或装置。数字机器依靠程序编程，与 PLC 关联很大。

数字控制允许操作员通过一系列的数字和符号与机床通信。一组指令构成数字控制程序，指挥机床定向切割部件、选择不同的刀具、控制切割速度和控制转轴旋转，还有一些其他附加功能，如控制冷却液流通或停止等。有多种语言可以用来编写数字控制程序，最常用的是自动程序设计工具（APTs）。

图 15-24 所示为典型的数字控制系统。控制器读取并解释指令，控制机器按照操作员期望的方式动作。当需要增加材料或更换刀具时，机床操作员会得到提示。系统控制器通常是一个工业计算机，它存储并读取程序，把程序信息转换为电气信号去驱动电机，实现对机床的控制。控制器提供信号给电动或液压马达，驱动机床头左右或上下移动。电机驱动工作台使工作部件前后移动。位置编码器和速度传感器给控制器提供反馈信号，这样就能知道

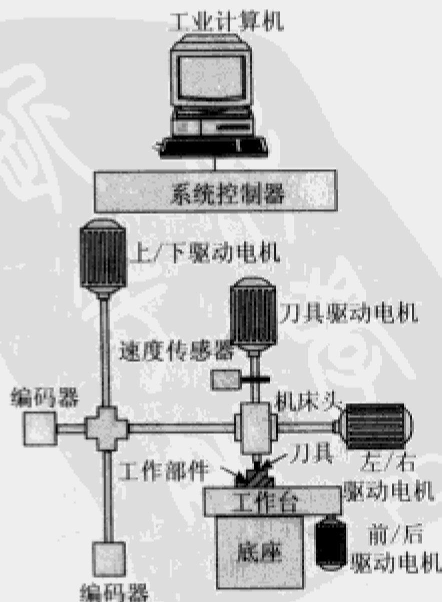


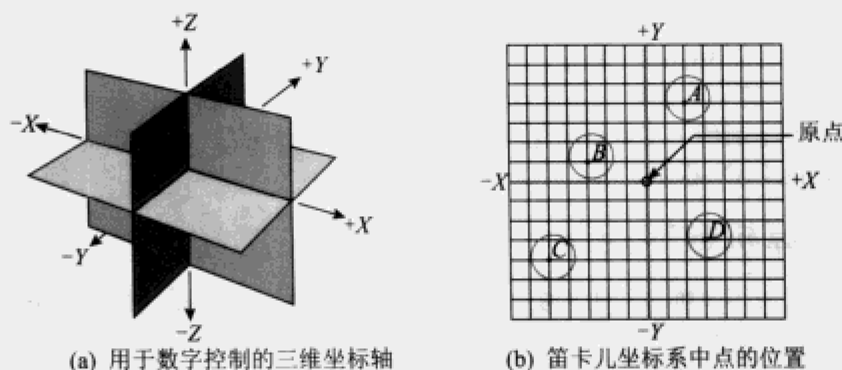
图 15-24 典型的数字控制系统 426

机床头的精确位置和速度。由于系统电子设备的精确度比机械精确度高，所以生产部件的重复性和一致性都在允许的误差范围内。

数字控制，非常适合于生产来自只有大小和外形不同的相似给料（原材料）的产品部件。即使产品数量比较少，数字控制也可以是经济可行的方案，前提是操作次序可以在同一个数控机床上完成。但是，当制造部件涉及多个不相似的操作次序时，在生产中就要使用多个数控机床。

如果控制机床是空间坐标类型，刀具的位置地址可以用字母 x、y 和 z 确定。使用 x、y 和 z 坐标，机床可以被控制定位到正确的位置。工作部件的位置，可以用笛卡尔（直角）坐标系确定。在这种坐标系中，一个点的位置可以被与互成直角的一组坐标轴的参考位置确定，如图 15-25 所示。垂直的轴是 y 轴，水平的轴是 x 轴。两个轴相交的点被称为零点或原点。x 轴原点左边和 y 轴原点下面，坐标值要在前面加负号（－）；x 轴原点右边和 y 轴原点上面，坐标值要在前面加正号（＋）。z 轴方向的运动与机床转轴平行，它是工作部件与机床之间的距离。

427



(a) 用于数字控制的三维坐标轴　(b) 笛卡儿坐标系中点的位置

图 15-25　笛卡儿坐标系统中点的确定

图 15-25b 中，用 x、y 和 z 确定机床运动位置的坐标。在笛卡尔坐标系中，每个点可以确定如下：

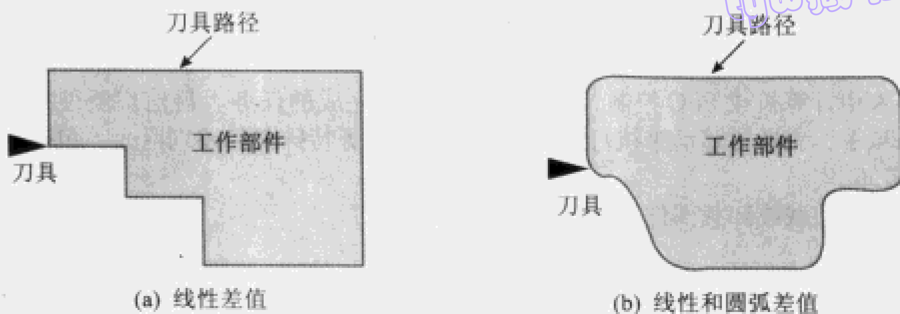
$$A(x=+2, y=+4)$$
$$B(x=-3, y=+1)$$
$$C(x=-5, y=-4)$$
$$D(x=+3, y=-3)$$

数控机床的编程可以分为两种主要形式：点对点编程和轮廓编程。点对点编程涉及直线运动。图 15-26 所示为对钻 4 个孔操作的点对点编程。每个孔的位置由 x 和 y 坐标值确定。钻完一个孔后，机床移动到下一个要钻孔的位置，顺序钻孔直到程序完成。机床在各个孔之间移动的路径不重要，因为刀具是在孔之间上方的空中移动。每个孔的深度由 z 轴坐标决定。



图 15-26　点对点编程

轮廓（也称为连续路径）编程，像在磨床上面加工部件一样，当刀削工具从一个工作点移动到下一个时，它始终与工作部件接触在一起。连续路径的定位，需要同时控制在两个或多个轴上的运动，以保持刀具与工作部件的相对位置关系不变。描述机床从一个工作点移动到下一个工作点的轮廓的方法称为差值。所有的轮廓控制都提供线性差值，大部分都有线性和圆弧差值（如图 15-27 所示）。



(a) 线性差值 (b) 线性和圆弧差值

图 15-27 轮廓编程

计算机数字控制（CNC），可以取代原来数控设备的穿孔带和硬接线式机器控制（如图 15-28 所示）。CRT 显示器显示当部件被加工时，机床工作台和/或刀具的准确位置。计算机数字控制（CNC）增强了制造工业的灵活性，因为基于微处理器的控制设备给数控机床带来了新的特征：

428

- 提高了程序存储容量，硬盘取代了以前的穿孔带。
- 简化了程序编辑。
- 借助计算机数学处理能力，能绘制更加复杂的部件轮廓。
- 可重复使用的机器模型，能根据需要存储和恢复。
- 通过许多外围设备，能实现全厂范围的通信。

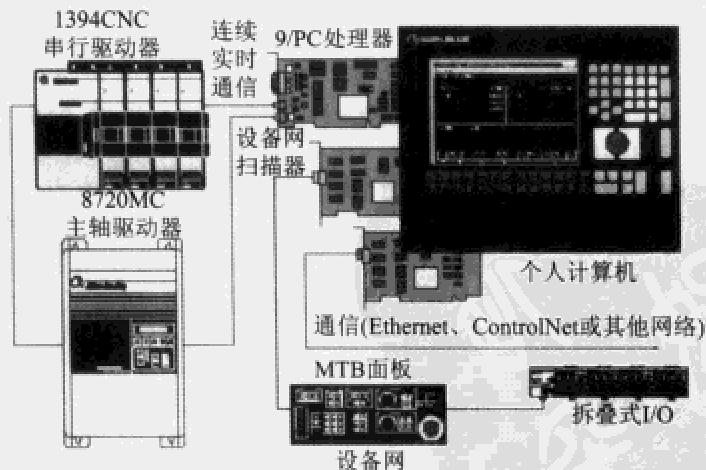


图 15-28 计算机数字控制（得到 Allen-Bradley 公司许可）

简单的计算机数控（CNC）编程包括从部件绘图中提取信息，然后把信息转换为计算机程序。可以在个人计算机上设计程序。用数字控制编程语言设计程序简化了程序编写，这是因为计算机可以计算出机器坐标。使用计算机辅助编程可以使程序做到如下几点：

- 存储在大容量存储系统中。
- 根据需要恢复和编辑。
- 在使用程序控制设备前进行离线调试。
- 用连接到计算机的绘图机绘图，帮助程序调试。

15.5 机器人技术

机器人由计算机或PLC控制，执行通常由人来完成的任务。目前广泛使用的工业机器人是机械手，它通过移动来执行工业操作。操作任务特殊并且差别巨大，包括：

429

- **装卸**
在机器上装载或卸载部件。
- **处理**
切削、钻孔、喷漆、镀膜。
- **装配**
把部件放置和固定在另一个部件上。
- **拆卸**
把设备分解成小的部件。
- **焊接**
通过电弧焊或点焊固定设备。
- **传送**
移动材料或元件。
- **喷漆**
给元件喷漆。
- **危险作业**
工作在高温、灰尘、放射性、噪声和有毒环境下。

机器人就是通过伺服电机驱动的一系列的机械连杆。连杆之间的接点称为关节或轴。轴有直线的（线性）、圆形的（旋转）或球形的。图15-29所示为一个六轴的机器人手臂。机械手臂的最后3个关节被称为机械腕。沿着手臂的腕关节分别是：斜关节、偏关节和转动关节。高性能的机器人有6到9个轴，随着性能增加，轴可以增加到16个或更多。这些机器人能模仿人类的动作。

机器人能触及的区域被称为工作包络。机器人触及范围内的所有点，都是工作包络的一部分。工作包络的外形由机器人的长轴类型决定（如图15-30所示）。包含两个线性长轴和一个圆形长轴的机器人，具有圆柱形工作包络。而包含三个圆形长轴的机器人的工作包络，就像是人的身体从腰部到肩部再到肘的活动范围一样。知道了机器人的工作包络，可以帮助避免人身伤害和潜在的设备损坏。

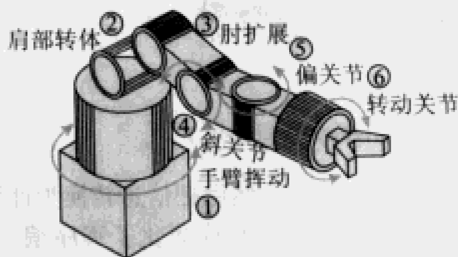


图15-29 六轴机器人手臂

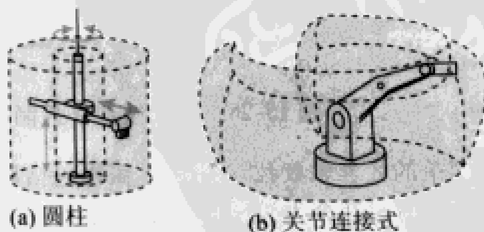


图15-30 机器人工作包络

430

对于大多数的操作，都需要连接附加的工具到机器人。机械臂末端的工具一般被称为末端执行器，末端执行器根据机器人的工作类型而异。夹钳用于设备装卸和装配。点焊和弧焊都配备自己的工具，喷漆和分配也一样（如图15-31所示）。

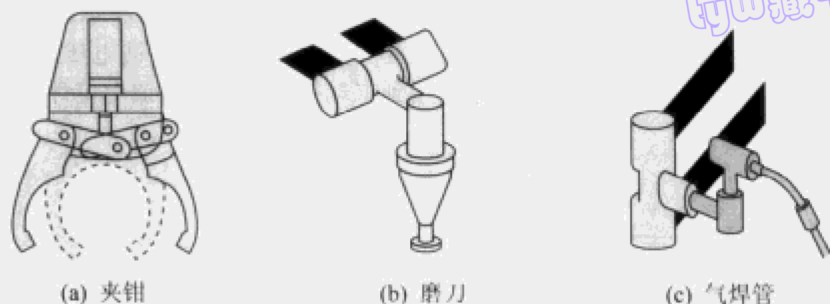
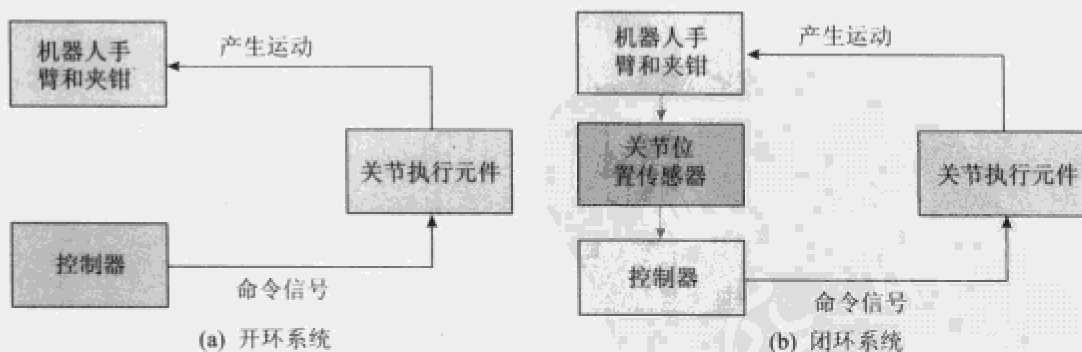

(a) 夹钳　(b) 磨刀　(c) 气焊管

图 15-31　末端执行设备

机器人通常具有下列 3 种动力源之一：电机、液压传动器（通过压力油驱动活塞）、气压传动装置（通过压缩空气驱动活塞）。通过压缩空气驱动的机器人重量轻、价格低、移动快，但是不够坚固。通过液压液体驱动的机器人更加坚固，价格也贵，但是当液压液体温度改变时，准确性会降低。

最初所有的机器人都是通过液压伺服驱动的。在早期的工业机器人中，大多数驱动的标准都只是为了维持液压伺服系统的稳定，后来工程师发展了用直流伺服电机驱动的多关节型机器人。工业机器人又从直流电驱动演变为交流电驱动，交流伺服电机比直流伺服电机有更大的优势。交流伺服电机是无刷、免维护性设计，并通过增量式编码器提供伺服位置反馈。

机器人控制系统有两种类型：闭环形式和开环形式（如图 15-32 所示）。在开环系统中，没有传感器或反馈信号检测操作器实际响应命令信号的状况。在闭环系统中，从关节位置传感器接收反馈信号，控制器比较关节实际位置与编程设定值，然后发出命令信号减少或去除偏差。



(a) 开环系统　(b) 闭环系统

图 15-32　开环和闭环机器人控制系统

伺服机器人常常用于闭环（反馈）系统，非伺服机器人常用于开环（无反馈）系统。非伺服机器人比较小，为驱动轻负载而设计。它们的每个关节只有两个位置（打开和闭合），能够高速运行。由于它们工作是从一个位置移动到另一个位置，通过编程调整机械限位开关完成操作任务，又常常被称为继电器式机器人。非伺服机器人非常适合取/放操作，如从机器上面装载和卸载元件。非伺服机器人没有对速度进行控制，工作在冲击运动模式下。伺服机器人不同于非伺服机器人，它的运动比较平缓。在操作器从一个点运动到另一个点时，它可以控制每个连杆的速度、加速和减速。根据程序，通过机器人在工作时，对一些状态的测量，可以形成多个不同的运动分支。

伺服机器人的每个轴基本上就是一个闭环伺服控制系统。一个简单的伺服系统如图 15-33 所示。在这个例子中，伺服放大器用来放大命令电压值和反馈电压值的偏差信号。产生的偏差信号驱动伺服电机，伺服电机连接到末端执行设备和反馈电位计上面。

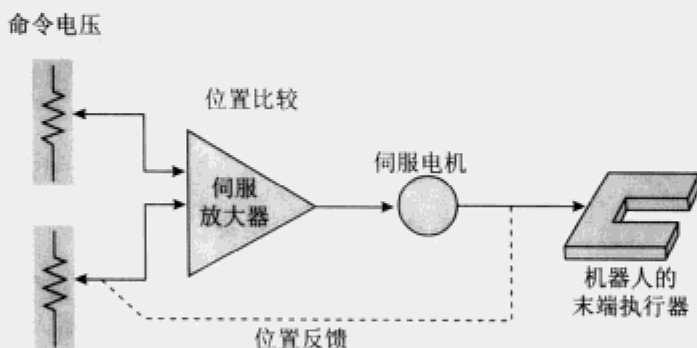


图 15-33 简单的伺服系统

控制器包含电源、操作员控制、控制电路和存储器。它指挥机器人的操作和运动，并控制与其他外部设备通信。它有下面三个主要的功能任务：

- 给操作器提供运动控制信号（也被称为信号处理）。
- 提供程序存储器。
- 解释包括操作指令在内的输入/输出信号。

控制器结构有不同的方式（如图 15-34 所示）。一般地，控制器包含下面的设备来控制系统：

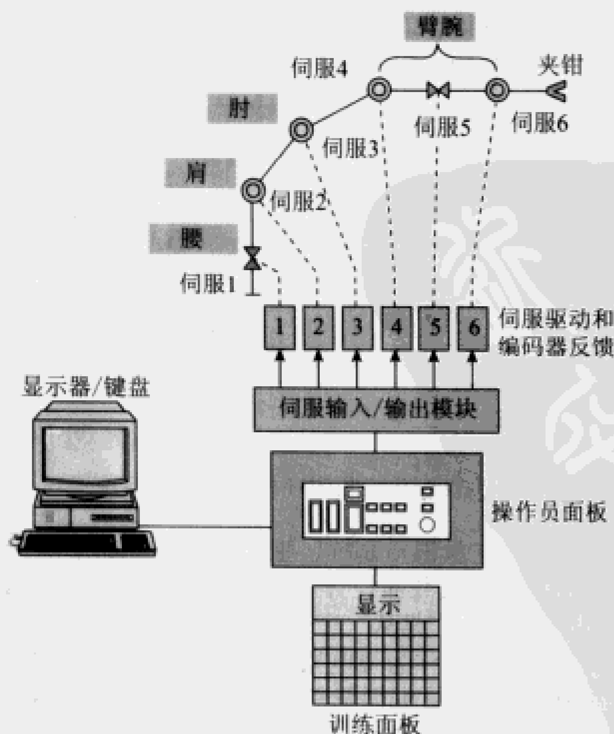


图 15-34 机器人控制系统

❑ **操作员面板**

装设有指示灯、按钮和按键开关。完成的功能：给系统上电/去电、校正机器人、故障发生后复位机器人、保持机器人的移动、开启自动化操作和紧急停止机器人。 432

❑ **训练面板**

装设有一个键盘和 LCD 显示屏，并通过插入式电缆连接到控制器内部的计算机 RAM 板上。有以下功能：微调机器人、训练位置数据、测试程序的执行和从错误中恢复；显示用户消息、错误消息、提示信息和菜单；显示和更改位置信息、程序变量和输入/输出；还执行一些与计算机文件有关的操作。

❑ **CRT 显示器和键盘**

如标准的计算机终端设备。有以下功能：执行计算机文件操作和显示状态和诊断信息，输入、编译和调试程序。

机器人控制器是基于微处理器的系统，它结合输入/输出卡或模块，一起完成操作任务。越来越多的机器人采用 PLC 控制。编程方式可以是线圈和触点，也可以是鼓轮/顺序器类型。在采用 PLC 对控制机器人进行编程之前，先要确定 PLC 与机器人连接的方案。

随着计算机和 PLC 在工业领域使用的增多，机器人控制器变得比它所控制的操作器更加重要。如今机器人控制器需要与外面的设备进行通信，如 PLC 或工厂的计算机系统。

实际上，所有的机器人都采用一些编程语言进行编程。这些编程语言控制机器人移动到特定位置、输出信号并读取输入信号。编程语言增加了机器人的灵活性，允许机器人反作用于外部环境（使用传感器）并做决策。机器人编程语言有很多种，目前对机器人编程语言没有统一的标准，每个厂家都有适合自己机器人的专用编程语言。

交替编程方法被称为 walk-though，也可以使用 teach-through 编程方法。在 walk-through 训练中，编程者控制操作器，按照它要学习的方法机械的移动操作器。控制器记录下移动轨迹，以备后面重演，速度可以很快。Teach-through 编程，使用操控杆或训练面板指导机器人按照预先设定的路径移动。控制器利用计算机教学软件，为机器人计算出平滑的路径。 433

思考题

1. 解释下面每个计算机硬件部分的功能：
   a. 电源　b. 软驱　c. 硬盘驱动器　d. 主板　e. 微处理器　f. ROM
   g. RAM　h. 外围卡　i. 扩展槽
2. 个人计算机通常采用哪些类型的硬盘驱动器？
3. 列写出 5 种装卸软盘驱动器的预防措施。
4. 比较硬盘驱动器和软盘驱动器的存储容量和存储速度。
5. 怎样辨别计算机中不同的磁盘驱动器？
6. 比较系统软件和应用软件。
7. 列写出操作系统的 5 种功能任务。
8. 描述出下面每种类型文件的内容：
   a. 可执行文件　b. 数据文件　c. 文本文件　d. 图形文件
9. 解释计算机 BIOS 的功能。
10. “系统引导”指的是什么？
11. 解释文件夹或目录的功能。
12. 解释 Windows 系统是怎样允许多任务操作的。
13. 列写出计算机上的 10 种类型的端口。
14. 描述出工业工作单元的组成。

15. 列写出3种类型的局域网（LAN）拓扑结构。
16. 解释协议对设备之间进行通信的重要性。
434
17. 比较主－从通信方式和对等通信方式。
18. 比较专有通信网络和开放式通信网络。
19. 列写出5种类型的开放式通信网络。
20. 讨论采用DeviceNet网络连接输入/输出设备的优势。
21. 简单描述出数字控制设备的工作原理？
22. 数字控制最适合于哪种类型的工业应用？
23. 解释在数控（NC）编程中，笛卡儿（直角）坐标系的功能。
24. 比较数控点对点编程和轮廓编程的异同。
25. 简单的计算机数字控制一般由哪儿部分组成？
26. 列写出由工业机器人控制的5种常见的操作任务。
27. 简单描述出机器人的组成。
28. 给机器人工作包络下个定义。
29. 列写出3种驱动机器人操作器的动力源。
30. 机器人系统控制器的主要任务是什么？
31. 写出机器人的3种编程方法。

习题

1. 确定你身边计算机的下列技术规格：
   a. 软驱的数目　b. 磁盘驱动器类型　c. 每个驱动器的标记符号　d. 每个驱动器的存储容量
435
2. 撰写一份关于一种类型的开放式通信网络的报告，利用互联网和制造厂家的资料帮助你完成报告。

Programmable Logic Controllers
Third Edition

PLC教程（第3版）

“我一直从事PLC的教学工作，这本书堪称完美！”
——Amazon.com

本书原版是在国外广受欢迎的一本教材，自出版以来其内容随着PLC技术的发展而不断完善，每一版都涵盖了PLC的最新功能，体现了PLC的最新发展趋势，因此在世界范围内均非常畅销。本书并不拘泥于某种型号PLC的具体细节，而是从PLC的共性特征出发，重点讲述PLC的基本工作原理，并提供如何安装、设计和维护PLC控制系统的实用信息。

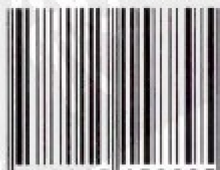
本书适合作为大学本科电气信息类及相关专业的PLC教材，也适合从事PLC应用和开发的工程技术人员。

本书特色

- 以通用的方法介绍PLC功能，内容适用于所有型号的PLC
- 语言通俗易懂，并配有大量清晰的图表，便于理解
- 提供故障诊断步骤，指导读者对PLC故障进行系统分析
- 每章均配有思考题和习题，便于读者巩固所学内容

McGraw Hill Education
http://www.mheducation.com





ISBN 978-7-115-15889-5
9 787115 158895 >
ISBN 978-7-115-15889-5/TN
定价：45.00 元

本书相关信息请访问：图灵网站 http://www.turingbook.com
读者/作者热线：(010) 88593802
反馈/投稿/推荐信箱：contact@turingbook.com
上架建议 电子电气/PLC
人民邮电出版社网址 www.ptpress.com.cn